

# SINC-LINK

Vol.8 No.2 Mar-Apr '90

SINC-LINK IS A PUBLICATION OF THE TORONTO TIMEX-SINCLAIR USERS CLUB AND IS ISSUED 6 TIMES A YEAR. COPIES OF THE NEWSLETTER ARE \$1.50 EACH FOR NON-MEMBERS. CLUB MEMBERS RECEIVE FREE COPIES AS PART OF THE \$20.00 ANNUAL MEMBERSHIP FEE. A NEWSLETTER SUBSCRIPTION ONLY IS AVAILABLE FOR \$12.00.

NEWSLETTERS ARE EXCHANGED, FREE OF CHARGE, WITH OTHER TIMEX-SINCLAIR USERS GROUPS.

PLEASE CREDIT THIS PUBLICATION AND THE AUTHOR IF YOU COPY MATERIAL.

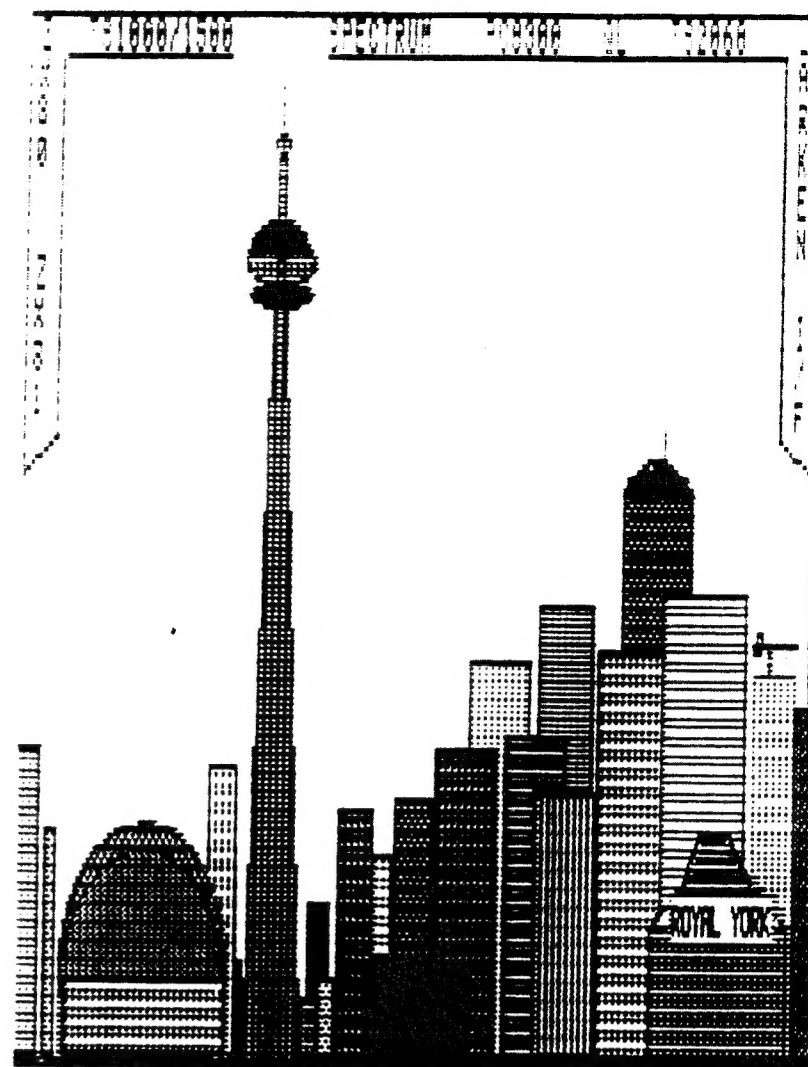
THE CLUB MEETS ON THE FIRST WEDNESDAY OF EACH MONTH AT FOREST HILLS COLLEGIATE INSTITUTE, 730 EGLINTON AVE. W., TORONTO. START TIME: 7:00 PM.

SEND CORRESPONDANCE TO:

Attention: SINC-LINK EDITOR  
TORONTO TIMEX-SINCLAIR USERS  
CLUB, 14 RICHOME COURT,  
SCARBOROUGH, ONTARIO,  
CANADA M1K 2Y1

## EXECUTIVE OFFICERS:

PRESIDENT:	RENE BRUNEAU ( 531-9749 )
TREASURER:	BILL LAWSON ( 444-8772 )
SECRETARY:	GEORGE CHAMBERS ( 751-7559 )
ACTIVITIES:	RENE BRUNEAU ( 531-9749 )
TAPE LIBRARIAN ZX81:	LYMAN PAQUETTE ( 482-4479 )
TAPE LIBRARIAN 2068:	RENATO ZANNESE ( 635-6536 )
TAPE LIBRARIAN QL:	HUGH HOWIE ( 634-4929 )
PAPER LIBRARIAN:	JEFF TAYLOR ( 244-8583 )
NEWSLETTER:	JEFF TAYLOR ( 244-8583 )
LIAISON OFFICER:	GEORGE CHAMBERS, 14 RICHOME COURT,
( Out-of-town members )	SCARBOROUGH, ONTARIO, M1K 2Y1
	( 416-751-7559 )



TORONTO TIMEX-SINCLAIR  
USERS CLUB

SPRING  
AHEAD  
ISSUE

DEDICATED TO  
TAKING YOUR  
MIND OFF THESE  
TAXING TIMES

TORONTO TIMEX-SINCLAIR USERS CLUB

## Editorial

Ready for a couple of scary words? Hardware projects! Doesn't that send a shiver down your spine? Does the thought of taking a red hot soldering iron to the innards of your precious computer make you break out in a cold sweat? Good.

Now that I've had some fun, let me point out that ordinary people just like yourself are busy soldering and snipping and building hardware projects for Timex-Sinclair computers every day and nothing awful happens to them. Unless you think that gaining a little knowledge or enjoying that glow of satisfaction you get from creating something that works is so awful.

Just look at yours truly. I am hardly what you'd call a computer whiz. I'm really just a user. I can barely program in BASIC, know no machine code, don't understand how the computer works and think that electronics are black magic. But, I am curious as to what my computers are capable of. If I can attach a gizmo that allows me to talk to other computers at 1200 baud or enhance the quality of my video display or expand the storage capacity of my computer then I think that's really neat. I'm just an average user who has a lot of curiosity. Am I so different from you? I don't think so.

To date I have added a keyboard and an internal static RAM to my TS1000, built a serial interface cable and an RGB buffer for my QL, and assembled an RS232 interface and an internal RGB buffer for my TS2068. And I'm working on a video digitiser right now. Why? Because all these hardware projects improve the capabilities of my computers and who doesn't want that?

In the next few issues we'll

be running articles about projects we *think* you'll be interested in. Is there something you'd especially like to see? Let us know. Have you built something that you'd like to share your experience with us? We'd like to hear from you. After all this is *your* newsletter, so make *your* interests known.

Hope to hear from you. That's all for now...

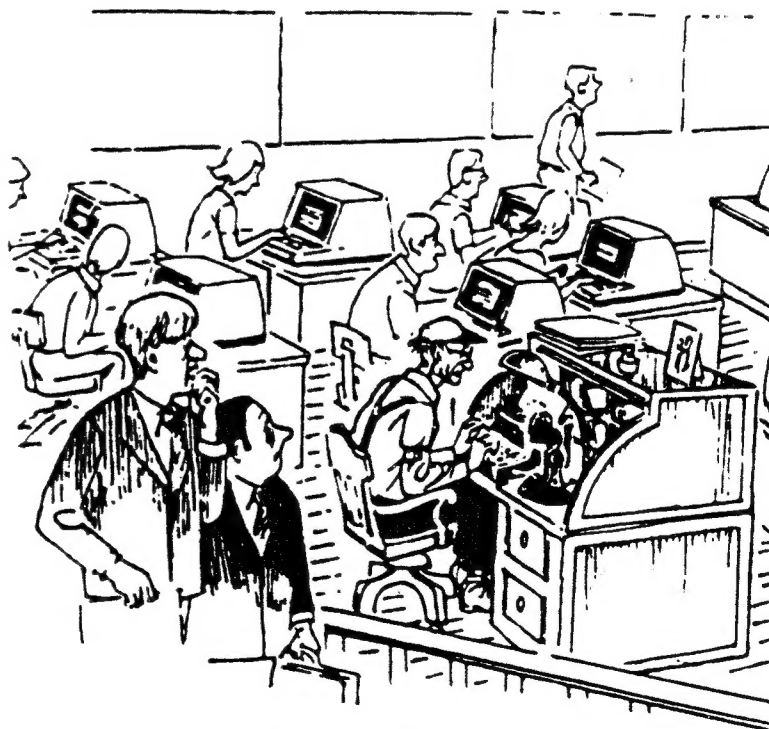
J.T.

FOR SALE FOR SALE FOR SALE

1 ZX81 Computer, with P/S,  
16K RAM, cords, tapes, books, etc.

Best offer

Call S. Speirs (416 485 5496)  
37 Donegall Drive,  
Toronto, Ont M4G 3G6



*"But we can't phase him out.  
He's the only one who can spell."*

SIMPKINS IN WORLD PRESS REVIEW

## BOB'S NOTEBOOK

Here is a utility for the Larken DOS which will read the header information for each file on a disk and give you the first track number, start address, the length, and in the case of BASIC programs, the autostart line number.

I usually make a copy of this information for each disk and tuck it inside the sleeve; it comes in very handy when you need to know this data and you have a program loaded that you are working on. For example, you might be putting in a line to save a particular code file and have to indicate the start address and length. Finding it on the slip inside the sleeve will save some time and frustration.

The first listing is the boot file to load and control the code. The second listing is the BASIC ready for the compiled program you will make using Timachine. The program was first written by George Chambers and I have made some minor adjustments to provide space for track numbers with three digits as with Quad Drives and to show the word "none" when there is no autostart line number in a BASIC program.

Type both of these in carefully and save them a couple of times before doing the compilation. I have included some notes after some lines; of course, these are not to be typed in.

### Listing #1 (Header Reader Boot File)

### :COMMENTS

10 REM reader boot. Designed for use with wide printer.	
50 RANDOMIZE USR 100: LOAD "reader.Cc" CODE 40000	:load compiled code
100 OUT 127,15	:set condensed mode
110 RANDOMIZE USR 100: OPEN #4,"dd"	:set LKDOS PRINT #4 command
120 PRINT #4: CLOSE #3	:close LKDOS stream #3
130 PRINT #4: OPEN #5,"lp"	:open stream #5 to wide printer
140 PRINT #4: POKE 16090,39	:set width of line to 40
150 PRINT #4: POKE 16094,8	:set left margin at 8
160 INPUT "drive? (0-3) ";drv	:pick a drive
170 PRINT #4: GO TO drv	
180 RANDOMIZE USR 40000	:activate header reader code
200 INPUT "another? (1=yes 0=no) ";an	:do another or quit?
210 IF an THEN GO TO 160	
220 IF NOT an THEN PRINT #4: GO TO 4: PRINT #4: NEW	:if NO then load RAMDISK autostart file (see note)*
300 PRINT #4: SAVE "reader.B1" LINE 1	:GO here to save.

\*If you want to quit to some other drive number, just adapt this line to your needs. Just remember:  
PRINT #4: NEW means load the autostart in the disk drive selected.

### Listing #2 (Header Reader File ready for Compilation)

90 REM !USR 40000	:forces compilation to start at add. 40000
100 REM ! OPEN #	:Timachine Command
170 RESTORE 210	
180 FOR n=VAL "63000" TO VAL "63077"	:pokes code in lines 210-280
190 READ a: POKE n,a	:into addresses 63000 to 63077
200 NEXT n	
210 DATA 195,43,246,195,72,246,195,104,246,243	:this code loads/saves directory
220 DATA 205,98,0,201,58,100,0,251,201,205	:track contents from disk to
230 DATA 33,246,58,176,92,50,29,32,205,126	:buffer area at 57500
240 DATA 0,205,123,0,33,112,32,17,156,224	
250 DATA 1,0,20,237,176,195,38,246,205,33	
260 DATA 246,58,176,92,50,29,32,33,156,224	
270 DATA 17,112,32,1,0,20,237,176,205,150	
280 DATA 0,205,126,0,205,120,0,195,38,246	
290 DIM c\$(448)	:array to make 14 blank lines
300 DIM i(100)	:array to store 100 numbers
310 DIM d\$(100,10)	:array to store 100 names
320 LET trac=23728: LET loadbuf=63000	
330 LET oo=0: LET oa=1: LET ob=2: LET oc=3:	:some variables to save memory

LET oo=4: LET oe=5: LET of=6: LET og=7:

LET f=oo: LET z=oo

340 LET name=57688: LET buffer=57500

350 BORDER oa: PAPER oa: CLS

360 PRINT AT ob,oe: INK og: PAPER oa: " LARKEN DISK  
UTILITY "; AT od,ob: "Header Reader by G. Chambers"; AT  
of,od: "Compiled by Bob Mitchell"

370 INK og: PLOT oo,VAL "108": DRAW oo,VAL

"62": DRAW VAL "255",oo: DRAW oo,VAL "-62":

DRAW VAL "-255",oo

380 INK og: PLOT VAL "8",VAL "114": DRAW oo,VAL

"50": DRAW VAL "238",oo: DRAW oo,VAL "-50":

DRAW VAL "-238",oo: INK og

385 INPUT "printout? y/n "; LINE y\$

386 LET s=2

387 IF y\$="y" OR y\$="Y" THEN INPUT "3=TS2040 5=WIDE "; s

388 PRINT #s

390 INPUT "Enter No. & Date (max 14) " LINE g\$:

IF LEN g\$>14 THEN LET g\$=g\$( TO 14)

395 PRINT #0: "Place disk to be read in correct drive

then press a key": PAUSE oo

400 REM Get names from Trk 0

410 POKE trac,oo: RANDOMIZE USR loadbuf

420 PRINT AT of+oc,oo: c\$( TO 320)

440 FOR n=oa TO 100

450 LET e=n

460 IF PEEK (name+oc)=oo THEN LET e=n-oa: LET n=100:

GO TO 530

470 IF PEEK (name+oa)=254 THEN LET name=name+34:

GO TO 460

480 FOR m=oa TO 9

490 LET d\$(n,m)=CHR\$ PEEK (name+m)

500 NEXT m

510 LET l(n)=PEEK (name+11)

520 LET name=name+34

530 NEXT n

540 CLS : PRINT #s: "DISK HEADER DATA: "; g\$

545 PRINT #s: "-----"

550 PRINT #s: "TRACK NAME    START    LENGTH S/Ln"

570 PRINT #s

590 PRINT #0: "(M returns to menu)": PRINT AT oc,oo

620 FOR t=oa TO e:

630 IF PEEK 23689<og THEN GO TO 830

650 POKE trac,l(t): RANDOMIZE USR loadbuf

670 IF PEEK (buffer+oa)<>oo THEN

680 IF PEEK (buffer+oa)=oo THEN GO TO 800

685 PRINT #s: PEEK (buffer+oa); TAB od;

690 FOR a=(buffer+ob) TO (buffer+10)

700 PRINT #s: CHR\$ PEEK a;

710 NEXT a

720 LET start=oo: FOR a=(buffer+ob) TO (buffer+10):

IF PEEK a=46 AND (PEEK (a+oa)=67 OR PEEK (a+oa)=65)

THEN LET start=oa

730 NEXT a

740 PRINT #s: TAB 14: PEEK (buffer+12)+256\*PEEK

(buffer+13);

750 PRINT #s: TAB 21: PEEK (buffer+22)+256\*PEEK

(buffer+23);

760 LET len=PEEK (buffer+17)+256\*PEEK (buffer+18)+oa

770 IF len=65535 THEN LET len=-1

:buffer starts @ 57500; file names 188 bytes later

:variable <s> is display device

:if s=2, data goes to screen

:if s=3, to TS2040; if s=5, to wide printer

:put in I.D.

:chooses trk 0 and loads it to buffer

:prints 10 blank lines

:counts names found

:if PEEK=0 then search ends

:if PEEK=254 then look for next name

:this loop gets the name

:gets first track number

:S/Ln=Autostart Line #

:this loop loads first track of each file name

:from each file to get required data

:prints track #

:then the prog. name

:identifies BASIC files as "0"

:others as "1"

:prints start address

:then file length

:gets the autostart line

:makes it -1 if 65535



```

780 IF start=00 AND len<>-1 THEN PRINT #s;TAB 27;len:      :if a BASIC file with an autostart line
GO TO 800                                                  :then print line #
785 IF start=00 AND len=-1 THEN PRINT #s;TAB 27;"none":    :if no autostart line, print "none"
GO TO 800
790 PRINT #s
800 IF INKEY$="M" OR INKEY$="M" THEN STOP                  :if menu called, STOP; ie, BREAK into BASIC
810 NEXT t
820 PRINT TAB od; PAPER ob;"Last file on the disk":       :finally, some on-screen prompts
LET f=oa
825 IF s<>2 THEN PRINT #s;"'"
830 IF z=00 AND NOT f THEN PRINT AT 19,oa;" Press any
key to clear screen"
840 IF z=00 AND INKEY$="" THEN GO TO 840
860 IF INKEY$="M" OR INKEY$="M" THEN STOP
870 PRINT AT od,oa;c$;AT og,oa;c$: PRINT AT oc,oa
880 GO TO 630
1010 IF INKEY$<>"M" OR INKEY$<>"M" THEN CLS :
PRINT AT og,oa;"Press any key for next record":
PAUSE 0: GO TO 350
8900 REM ! CLOSE #                                         :Timachine Command
9000 RANDOMIZE USR 100: SAVE "reader.Bp"                   :GO here to SAVE.

```

#### CHANGING THE NAME OF A DISK

When you format a new disk, you can assign a name to the disk. But it is impossible to change that name after the disk has been formatted unless you reformat the disk and all the data on this disk will be lost.

The following program ( by Real Gagnon of **QL\_DOC** ) will correct this situation. You can now rename a disk without having to reformat the disk.

When you run the program, you will be asked for the number of the disk drive ( e.g. 1 , 2 , 3 or 4 ) . This program will only work with disk " " floppies " " and you cannot rename microdrives. The program will then display the name of the disk and will ask you for the proposed name. The limit is 10 characters.

As usual a copy of the program will be available from the librarian H. Howie.

Louis Laferriere

```

100 REMark Disk_RE_Name
110 REMark by Real Gagnon
120 :
130 REPEAT loop
140 PRINT #0,"Disk #. "!:
145 d$=INKEY$(-1)
150 IF d$ INSTR '123456789':PRINT #0,d$: EXIT loop
160 END REPEAT loop
165 :
170 OPEN#10,"flp"&d$&"_D2D"
180 :
190 GET#10\1,a$
200 PRINT#0,"Present name :"!a$(5 TO 14)
210 INPUT#0,"New name :"!n$
220 LET a$(5 TO 15)=n$
230 PUT#10\1,a$
240 :
250 CLOSE#10

```

Q L I P S

We have all had that "crashing" experience with the QL, and I wonder how often the crash is our own fault? My own little beast of burden had a problem at one time, but after some operation on its innards, the crashing was not quite so frequent, until recently that is, when I installed a plastic Carpet saver under my chair. This, combined with our present day mixture of wool and man-made fabrics, both in our upholstery and our clothing, made an ideal breeding ground for STATIC.

The problem seemed to arise when I used the printer, I had also noticed that when I left the Work Station, and returned, I was often welcomed with a little jolt to my fingers as I approached the QL and/or my Music Centre which is within reach of the QL.

The solution?

I use a sheet of thick corrugated card-board on top of my desk, on which my equipment rests, (it is easier on the arms than wood). So I went to the hard-ware store and bought a roll of Aluminum Tape, about 1 1/2 inches wide, the kind used in duct-work. Now don't get the fabric type, get the METAL type. I placed a strip along the front of my desk, about an inch from the edge and along the end, to the back of my desk. From there, I attached a piece of wire, which goes to the casing of my disk power supply, which is the nearest ground I could get. To attach the wire to the aluminum strip, I bare an end of wire about two inches back, make this into a loose coil, lay it on top of the card-board, and place the tape on top of the coil. The other end is attached to the power supply by using one of the screws on the cover.

CAUTION:- Ensure that the aluminum tape is cleaned a little at this point, as the adhesive is not always conductive material.

Next step is to get some Scotch Tape, about 3/4 inch wide, and lay a strip of this along both edges of the aluminum tape, ensuring you do not cover up too much of the width of the aluminum tape. The reason for placing Scotch Tape along the edge of the aluminum tape, is to help prevent the edge of the aluminum tape from lifting, also, if the aluminum tape should wear out a bit in the centre, then there should always be a complete circuit under the Scotch Tape. Follow me ? ( Check continuity ) You must ensure there is an ample width of aluminum tape available for hand and arm contact, in the middle of the strip.

To finish this off, solder into the wire to ground, a one meg-ohm resistor. This will allow the static charge to bleed away slowly, thus eliminating the little jolt you might get when you touch the strip.

Now, when I approach my console, my hands MUST pass over or TOUCH this strip, grounding myself out. When I turn aside to the printer, I am always cautious to touch the strip, and also when I return to the computer, I again touch the strip. I do not have to consciously do this, as my hands have to cross this strip to reach ANY equipment on the desk.

(The least amount of movement can set up a static charge.)

The shiny appearance is a constant reminder of a static possibility.

It costs only a few dollars, but can save so much damage being done, to nerves and equipment.

H.H.H.

SINC-LINK

PENETRATOR - a TS2068 game  
Modifying it to the Larken system  
by G.Chambers

PENETRATOR is an older arcade game for the TS2068, that was put out by Timex in 1983. It is readily converted to Larken disk operation. Simply use a Header Reader to locate the program code; and inspect the front end Basic to find the start USR address.

However the program contains a Landscape Editor to allow you to modify the game. This Editor also allows you to save the modified landscapes code to tape, and to reload it. The challenge was to convert this feature to the Larken system.

The program code proved difficult to break out of, once the game was started. Often it is possible to break out of a m/c program by using the NMI-button and either pressing the A key or doing a SAVE to a protected disk. However, in this case every use of the NMI-button resulted in a system crash. This was going to make things difficult.

With Spectrum programs the first thing I usually do is to search for a CALL to a LOAD routine in the ROM. I do a FOR/NEXT loop looking for appearances of a 205, followed by an 86 and a 5. In the Spectrum parlance this could mean a CALL to address 1366, the start of the Spectrum LOAD routine. In this case it was a TS2068 program, so that meant complications since the SAVE/LOAD routines for the TS2068 are in the EXROM. I found that the program could be loaded, and would work in both the Spectrum and the TS2068 modes. This meant that the SAVE/LOAD routines were not in the ROM at all, but were imbedded in the program itself.

Since there was not likely to be any call to address 1366, and the location of the LOAD/SAVE routines were unknown, I then looked for a LD IX instruction (221) in the code, again using a FOR/NEXT loop. The LD IX instruction is used as a preface to most every SAVE and LOAD instruction. I wrote the FOR/NEXT loop so that every time it encountered an address containing a 221 it would print out a value based on the contents of the next two addresses. I multiplied contents of the higher-numbered address by 256 and added it to the contents of the lower address. If there was an LD IX used in a LOAD or SAVE routine this would give me a starting address. I'm looking for clues.

I found quite a number of promising addresses. I then loaded a disassembler at address 48000 (the disassembler "spec48.Cc" found on the club Larken disk #17). In loading, it overwrote part of the program code, but I guessed that the code we wanted to look at was below address 48000. If it had been higher I would have loaded "spec28.Cc" at 28000. I looked over the addresses and identified several promising ones. They contained sequences of LD A, 0 ; LD IX 47117; and LD DE, 7376. These are hot prospects for SAVE/LOAD m/c routines.

When I reloaded the program code, and then did a series of USR calls to these addresses I produced tape LOAD and tape SAVE effects. I explored the code further and found a USR call that would produce the screen instructions for the SAVE and LOAD actions. I was hot on the trail!

My first thought was to have the program exit to Basic to do the load and save routines from Basic. However I could not seem to exit gracefully from the program; my understanding of m/c programming leaves something to be desired! What I then decided to do was to employ a variation of a m/c routine that I had used earlier in the program SNODGITS. I would arrange that the program, instead of calling up the tape routines, would call up my own disk-save and disk-load routines. The PENETRATOR program code started at 32768, so I chose to locate my code at 32000.

Now, the disk routines require that a name be given to the program to be saved. What better than to make use of the name that one enters at the Editor portion of the program. This is fine, but now it was necessary to locate where the name was stored in the program code. I loaded the program, ran it, went into the landscape editor SAVE mode, entered a name, and just prior to the tape-SAVE I did an NMI-save. I had to do this because there was no way that I could break out of a running program. What I therefore planned to do was to inspect the program using the "doctor.B1" program ("doctor.B1" is a Larken utility on the club Larken disk #1).

Now, I had a fairly good idea of where the landscape name might be located. Earlier, when I did an experimental SAVE of a landscape, and had tried to read the header, I found that it was an unusual header. I won't go into details about it, except to say that I found the word "GARBAGE" in it. I therefore did a search for this word in the program, using the ASCII printout feature of the "spec48.Cc" disassembler program. I had located it at about address 47126, so I explored that area, with "doctor.B1", as I mentioned. And was successful.

I found there was space reserved for a 6-character name at address 47117. I decided to POKE a ".C1" in the three addresses immediately following the name. These addresses contained the start of the word "GARBAGE" I mentioned earlier. I reasoned that this was likely to be unused in the Larken mode. I found that the program would accept only 6-character names. That is to say, when the programs asked for a name it would only act on the first 6 characters. This, added to the ".C1" that I placed in the program would make up a proper Larken name.

I also went through the program and changed several instruction sequences which referred to TAPE, making them DISK. I ignored the tape verification routine, though it could have been deleted.

The disk LOAD/SAVE routine used in the SNODGIT program had to be modified to suit this particular application. I wrote separate disk SAVE and LOAD routines; and modified the PENETRATOR code addresses 46650 (LOAD), and 46843 (SAVE) to CALL the appropriate routine. Why those two addresses? Well, these were the addresses of the CALLs to the tape SAVE and LOAD routines, so I simply did a redirect to my disk routines.

Now, the Larken SAVE routine requires that the starting address and length of the code to be saved (and loaded) be specified. Because of the unusual nature of the tape SAVE headers it was not possible to determine this information



```

10 FOR n=32000 TO 32128
20 READ a: POKE n,a
50 NEXT n
100 DATA 195,0,128,0,0,0,205,30
,125,195,0,128,0,0,0
101 DATA 205,80,125,195,0,128,0
,0,0,0,0,0,0,0
102 DATA 243,0,205,98,0,33,13,1
84,17,34,32,1,10,0,237
103 DATA 176,62,11,50,2,32,205,
198,0,33,0,208,34,51,32
104 DATA 33,208,28,34,49,32,205
,201,0,58,100,0,251,201,0
105 DATA 0,0,0,0,0,0,243,0,205,
98,0,33,13,184,17
106 DATA 34,32,1,10,0,237,176,6
2,11,50,2,32,205,204,0
107 DATA 33,0,208,34,51,32,33,2
08,28,34,49,32,205,207,0
108 DATA 58,100,0,251,201,0,0,0
,0,0,0,0,0,0,0

```

47112	79	O
47113	82	R
47114	84	T
47115	41	)
47116	0	
47117	0	
47118	0	
47119	0	
47120	0	
47121	0	
47122	0	
47123	46	.
47124	67	C
47125	49	1
47126	28	
47127	71	G
47128	65	A
47129	82	R
47130	66	B
47131	65	A
47132	71	G
47133	69	E

46650	205
46651	6
46652	125
46653	48
46654	36
46843	205
46844	15
46845	125
46846	6
46847	50

from the tape headers. A close look at the tape SAVE routine in the PENETRATOR code showed a LD IX, 53248 and a LD DE, 7376. These could indicate a starting address 53248 and code length of 7376 bytes and seemed promising. The code contained other possible numbers, however the above were put into the 'disk' code, and proved to be correct.

The Disk code was arranged so that after a SAVE or a LOAD function was completed, a JUMP was made to address 32768, the start of the program. This was similar to the original program operation.

That was about it except to do a SAVE of the new code block 32000,33535 that included the modified program code plus the two disk routines.

\*\*\*\*\*

31999	00	NOP	
32000	C30080	JP	32768
32003	00	NOP	
32004	00	NOP	
32005	00	NOP	
32006	CD1E7D	CALL	32030
32009	C30080	JP	32768
32012	00	NOP	
32013	00	NOP	
32014	00	NOP	
32015	CD507D	CALL	32080
32018	C30080	JP	32768
32021	00	NOP	
32022	00	NOP	
32023	00	NOP	
32030	F3	DI	
32031	00	NOP	
32032	CD6200	CALL	98
32035	210DB8	LD	HL,47117
32038	112220	LD	DE,8226
32041	010A00	LD	BC,10
32044	EDB0	LDIR	
32046	3E0B	LD	A,11
32048	320220	LD	(8194),A
32051	CDC600	CALL	198
32054	2100D0	LD	HL,53248
32057	223320	LD	(8243),HL
32060	21D01C	LD	HL,7376
32063	223120	LD	(8241),HL
32066	CDC900	CALL	201
32069	3A6400	LD	A,(100)
32072	FB	EI	
32073	C9	RET	
32074	00	NOP	
32081	F3	DI	
32082	00	NOP	
32083	CD6200	CALL	98
32086	210DB8	LD	HL,47117
32089	112220	LD	DE,8226
32092	010A00	LD	BC,10
32095	EDB0	LDIR	
32097	3E0B	LD	A,11
32099	320220	LD	(8194),A
32102	CDCC00	CALL	204
32105	2100D0	LD	HL,53248
32108	223320	LD	(8243),HL
32111	21D01C	LD	HL,7376
32114	223120	LD	(8241),HL
32117	CDCF00	CALL	207
32120	3A6400	LD	A,(100)
32123	FB	EI	
32124	C9	RET	
32125	00	NOP	



# TRICKS OF THE TRADE

Left

Center

Right

by Mike Felerski

Anyone who uses a wordprocessing program has done it. Greeting card programs even do it. But what if I want to do it in my own programs?

If you are not lost yet, what I am speaking of is LEFT, CENTER, and RIGHT Justification of text lines on a screen display or on a hard copy printout. Left justification is when the first character of each line of text in a paragraph lines up evenly across the left margin of the screen or page. For example, this paragraph is left justified.

Center justification is placing the text line, headline or paragraph in the middle of the screen or page, like this paragraph.

Center justification is just placing the text line, headline or paragraph in the middle of the screen or page, like this paragraph.

The other evening I found that I needed some routines that would allow me to left, center, and right justify a string (t\$) of text for display. The three routines which are discussed here accomplish this by making a copy of the text string and then placing each character back into the original string in the new format (see Listing B).

```

10 REM Justify It
20 REM by Mike Felerski
30 LET max=32
40 DIM t$(max)
50 DIM o$(max)
60 DIM r$(32)
70 POKE 23658,8
100 GO SUB 300
110 LET t$=r$
112 PRINT AT 14,5;"(L)eft"
120 PRINT AT 15,5;"(C)enter"
130 PRINT AT 16,5;"(R)ight"
140 INPUT "Choice? ";c$
150 IF c$="L" THEN GO SUB 400
160 IF c$="C" THEN GO SUB 4100
170 IF c$="R" THEN GO SUB 4200
180 GO TO 140
200 REM Clear and Print
205 PRINT AT 8,0;"54321 9876543
2100123456789 12345"
210 PRINT AT 10,0;t$: RETURN
300 REM Enter String
310 PRINT AT 20,0;"Enter String
";max;" Chars Max"
320 INPUT r$: LET t$=r$: GO SUB
200: RETURN
    
```

Listing A

The routines assume that the text string is stored in t\$ and that the size of both t\$ and o\$ are DIMensioned to max. o\$ is used as the temporary string. Listing A is a demo/main routine that let us enter, justify, and re-display it.

```

4000 REM Left Justify
4005 LET o$=t$: LET pos=0
4010 IF o$(pos+1)=CHR$ 32 THEN
LET pos=pos+1: GO TO 4010
4020 IF pos=0 THEN RETURN
4030 FOR x=pos+1 TO max: LET t$(
1)=o$(x): LET i=i+1: NEXT x
4040 FOR x=1 TO max: LET t$(x)=C
HR$ 32: NEXT x
4050 RETURN
4100 REM Center Justify
4110 GO SUB 4000: LET o$=t$: LET
pos=max
4120 IF o$(pos)<>CHR$ 32 THEN G
O TO 4150
4130 LET pos=pos-1: IF pos=0 THE
N GO TO 4150
4140 GO TO 4120
4150 LET pos=(INT ((max-pos)/2)+
1)
4160 FOR x=1 TO pos-1: LET t$(x)
=CHR$ 32: NEXT x
4170 LET i=1
4180 FOR x=pos TO max: LET t$(x)
=o$(i): LET i=i+1: NEXT x
4199 RETURN
    
```

Listing B

Once the listings A, B, and C are typed in, RUN the program, enter some text, and then test the options to see the routines at work. If you are using ZEBRA SYSTEMS OS64, then try replacing line 30 with LET max=64. You may also replace the PRINT statement in line 210 with an LPRINT.

```

4200 REM Right Justify
4205 LET o$=t$: LET pos=max+1
4220 IF o$(pos-1)<>CHR$ 32 THEN
GO TO 4230
4222 LET pos=pos-1
4225 IF pos=0 THEN RETURN
4227 GO TO 4220
4230 LET i=max
4235 FOR x=pos-1 TO 1 STEP -1: L
ET t$(i)=o$(x): LET i=i-1: NEXT
x
4240 FOR x=i TO 1 STEP -1: LET t
$(x)=CHR$ 32: NEXT x
4250 RETURN
    
```

Listing C

The theory behind the routines is to find the first non-blank character in the string whose position is placed in pos. We look left to right, to left justify; and right to left, to right justify. This is the point at which we start to pick out each character and place them from o\$ into t\$, starting at t\$(1) or t\$(max) for left or right justification respectively.

In order to center the string, we first left justify the string, using a GOSUB 4000 so there is less guess work as to where the first non blank character is. Then we count back through the string, looking for the first non-blank character, starting from position max. We then take this value (pos), divide it by two and add 1 (adding 1 is optional per programmer's taste). This then gives us the starting lefthand position within the string. Finally the characters are picked and placed just as in the left and right routines.

Retyped from the I.S.T.U.G newsletter, by GFC (with minor revisions to the listings)

## MODEMS

About a year ago, it was mentioned that Modems were available at a very good price, and some members got them. Then in July this year, it was announced that more modems had been released by Bell. The first lot were priced at \$42, and the latter at \$67.50. The difference is that the latter lot at \$67.50, are Smart Modems, Hayes compatible, and are Auto Dial. So after seeing the demo Jeff Taylor put on, I was hooked, I just had to get one of these things. I sent away for it, and got it back two weeks later. Pretty good service.

The modem is 10" long 6" wide, and about 1 1/4 high. Nice light-grey plastic with darker cover. The lower lip sticks out about an inch, and this contains all the lights, as well as Pressure Membrane switches. There are 5 numbered switches to carry most-used numbers in memory. There are actually 30 memory cells in the modem. Inside, many flip-out chips for ease of repair if ever needed.

Installation was simple, or should have been. When I connected it all up, I could not get a dialling tone from the phone, I eventually found (with the assistance of Senen Racki), that the wall receptacle was more tolerant than the modem, to my phone plug. After changing that plug, I was in business. Such a small thing.

The modem operates either in TERMINAL or in COMPUTER mode. To switch from Terminal to Computer Mode, you have to flip off the cover, and change a jumper plug over. Takes longer to tell than it does to do.

In Terminal Mode the controls on the front are operative, but not in Computer Mode, as all the commands come from the keyboard. The lights of course perform their functions in all modes. I elected for Computer Mode.

I then loaded in "Q-Code Terminal" and proceeded to enter some numbers into the Directory, in anticipation of going On-Line. Of course the first time I use it I want to use the Auto Dial feature, but find that this does not work. I can Manual dial through the keyboard, and after I get the message CONNECT, I can press two periods, and get on line that way, but that is not Auto Dial. I can go through a whole bundle of trial and error, which is of little interest to anyone, what does matter is how I went on Auto, at least how I did it.

In our hurry to get involved we are inclined to toss the manual aside, not reading it properly, so after a few failures I went back to the manual, re-read it, and managed to work out what works. For me.

The answer would appear to be to go to the PASSWORDS, and I enter this for each number:-

--> ATDP#number..M <--

AT	Attention	(UPPER CASE IS A MUST)
D	Dial	(FOR ALL COMMANDS)
P	Pulse.	
#		
number	Number to be called	
..	Two periods	
M	CTRL+M	(Carriage return)

A lot of my initial trouble was I was using Lower Case.

When you go to Directory ( F3 D ) You select the number you wish to call, press 'T' and then 'F2'. You will now hear a lot of beeps clicks and clacks from the modem, dialling tone, and ringing tone, and if you are lucky a voice will come from the speaker in the modem, you then pick up the receiver and talk. Should you be calling a BBS, the word 'CONNECT' will flash on the screen, instructions will appear, and you don't have to touch the receiver.

Using on screen cursor select number from Directory, Press 'T' Press 'F2'

I have not yet tried to transfer files to or from as yet, as I have only had this for a week. The real reason for this missive is to let everyone know that there are still some of those modems left, and where to get them. Both models are available.

Rixon 212A data set for \$42.00 Plus post.

GDC 212A/ED+ for \$67.50 Plus post.

From Mr Croft B. Taylor  
Telecom Canada  
410 Laurier Ave. W.  
Room 410  
Ottawa. Ont. K1P 6H5

You will find Mr Taylor very helpful if you have a problem. Make remittance out to Croft B. Taylor and NOT to Telecom.

Hugh H. Howie.

## PRINTER PROSE

About three months ago I decided that I had been wishing for a printer for long enough, agonizing over whether I really NEEDED to spend that kind of money on my "hobby". (Sound familiar?) I finally talked myself (read 'my wife') into it and made the plunge. As a result, George Chambers asked me to describe my experience (so blame him for this).

The interface, a Tasman 'B' I think, was provided by a former club member through George. The first thing I did upon receipt, and before reading the instructions, was load the Tasman driver software and tried to LLIST the BASIC. On the first attempt, I got a single line of characters which had been overprinted many times. I read the printer manual and found a possible solution. It required a DIP switch adjustment. That done, the LLISTing worked fine.

I then attempted to print a screen COPY. Unfortunately, this time, each pass of the print head was separated by a line, I decided it was time to read the interface instructions.

Upon doing that I found I was able to specify the correct codes for Line Feed and Carriage Return for my printer quite easily by following the instructions in the BASIC program. I also had to reset the DIP switch! (The Moral is - READ THE INSTRUCTIONS FIRST!)

Having cleared the first two hurdles, I decided it was time to try out some of the fancy fonts of my Panasonic KX-P1180. It was possible, after a fashion, to do this from the control panel of the printer. However, I wanted the computer to do it! Therefore, I looked up the section on sending control codes to the printer in the manual and wrote a short BASIC program to print text in Enlarged mode.

You guessed it, it didn't work.

For several hours, over two nights, I tried, using both the Tasman driver and the V3 Larken driver, with no luck. I was stumped. I reread the interface manual, back issues of this newsletter and all my magazines, over and over until I was about to give up. Then I saw my mistake.

In all the literature, it stated to send an ESC character (CHR\$ 27) prior to sending the control sequence for the font required. However, as I finally discovered, the Tasman interface software is different. It requires an ESC prior to each code in the escape sequence in addition to the original ESC code. For example, with my printer, the code sequence given in the manual for enlarged text is ESC+W+1 in ASCII or CHR\$ 27; CHR\$ 87; CHR\$ 1 in decimal codes. To get this to work with the Tasman software, I had to send the sequence CHR\$ 27; CHR\$ 27; CHR\$ 87; CHR\$ 27; CHR\$ 1. Therefore, Moral #2 is - READ THE INSTRUCTIONS CAREFULLY.

I will now mention something else that may not be common knowledge about the Tasman interface. It will not work with the Larken Disk Interface. According to Larry Kenny, it has to do with the addressing of the Tasman interface. Apparently it is poorly buffered (forgive me if I get this wrong Larry) and

interferes with DOS commands. (I now use an Aerco type interface.) This leads me into the next part of the story.

Having to load the driver from tape each time was getting to be a nuisance so I decided to try using the Larken driver on the Version 3 EPROM. I read the Larken manual and found that it requires a POKE to set up the driver for the Tasman. That is, POKE 16096, 1. I also knew from my past trouble that I didn't want a Linefeed with the Carriage Return so this required a POKE 16092, 0.

I was now ready to roll. Or so I thought. I could send text to the printer all right but I still could not send the control codes with the Larken driver. This resulted in another period of study of back issues. I finally found the answer (but for the life of me I can't remember where) in an article for a different disk system. Apparently, the writer of the article was having the same problem I was and asked the designer what the trouble was. The solution? Send ASCII characters instead of SINCLAIR characters.

To do this with the Larken you have to bypass the routine that sends the SINCLAIR character set to the printer. This poke is found on page 9 of my Larken manual and is POKE 16093, 32. When this is done, no SINCLAIR tokens will be sent to the printer and ESC sequences will work. As a result, the code sequence for Enlarged text is CHR\$27; CHR\$ 87; CHR\$ 1. Incidentally, the Larken does not require the extra CHR\$ 27 required by the Tasman driver.

I am still puzzled by one thing however. I have not been able to send control codes using the OUT 127, CODE that I have since seen in programs published in this newsletter. I have checked the values for my computer (using the familiar PRINT IN 127) and found them to be 237 when ready and 233 when not ready. All I have gotten so far is widely spaced lines and characters, nothing like the effect I was trying for. I intend to keep on trying until I can figure it out but if someone already knows the answer, please enlighten me.

That about sums up my experience during the first few days with the printer. I hope this was a help to someone.

Lionel M. Keeping

## Update Magazine

1317 Stratford Ave.  
Panama City, FL 32404  
904 871 3556

## **NEWS AND NEW PRODUCTS**



\*\* That Tasword files may be loaded into Mscript? Assuming that both use the .CT extension and imbedded commands are not used the file will load easily. There will be some clean-up required since Tasword doesn't use carriage returns. Some words will end up run together and there will be extra spaces in places but it is relatively easy to fix with Mscripts 'insert space' key(1), 'delete' key(0) and 'delete left' key(9).

\*\* That Mscript files may be loaded into Tasword? The conversion isn't quite as easy as above, but it can be done. Mscript uses carriage returns (ENTER) and they will show up in Tasword as the graphic found on the '8' key. That is your clue for paragraphs and blank lines. If you work from the bottom of the document and come up the paragraphs can be separated using 'insert line' (SS-AND). The next step is to delete all leading spaces for any line. Then the paragraph can be reformat with 'reformat to end of paragraph' (SS-STEP). The paragraph will probably have some split words, delete extra spaces there and reformat again if necessary. (Taswords 'What-see-is-what-you-get' format makes the load into Mscript easier than vice-versa.)

\*\* That version 3 users can save a block of disk space when saving Artist II by adding the line:

1 BEEP 1,1: PAUSE 80

Enter GOTO 1 and press the NMI button when the long beep ends, press SHIFT and 1. Version 3 owners can then rename the file by

MOVE "NMI-S1.CM", "artist.B9"

or whatever name suits you. I use .Bn or .B9 to indicate NMI saves - the change from .C to .B extension avoids adding 'CODE'. I imagine there are other programs that can be done this way. It also gives you a clean bottom line when the load is completed.

\*\* If you load a program with machine code in a 0 REM statement that you shouldn't use the OPEN# 4, "dd" command? BytePower programs often use 0 REM code, so don't blame the program if you have used a 'boot' program with OPEN# 4. The solution is as simple as removing 'OPEN# 4, "dd" ' and replacing 'PRINT #4' with 'RANDOMIZE USR 100'. Merely trying to CLOSE# 4 doesn't resolve this problem. This one is in the Larken instructions but I forgot!) The lesson here is to reread the instructions and past SINC-LINKs every so often!)

\*\* That version 3 owners can make any program pause with the NMI button after inputting the following one liner:

RANDOMIZE USR 100: POKE 16100, 201:

RANDOMIZE USR 100: POKE 8214, 16100

The program pauses when you press the NMI button and may be resumed by pressing the "F" key.

\*\* That if you all share your tips in SINC-LINK we could all profit from it!

Les Cottrell

Cocoa, FL USA

\*\* that version 3 users can re-boot from within any program with the NMI-F feature? This assumes you have an AUTOSTART program written in Basic in your booting drive. After writing the pause routine earlier to stop a game with a count-down clock I started considering other ways to use this clever feature. I had been adding the GFC re-boot routine to my basic programs, but many MC programs still required turning off the computer. By installing the following routine in my RAMdisk boot program I can now re-boot from any program. The heart of this program is lifted from the instructions in the LARKEN Disk Editor instruction manual. Once the program is loaded I merely press the NMI button, then the "F" key and my RAMdisk is selected and booted. Obviously the program can be saved independently and run from your menu or as you choose.

The first number in line 5 determines which drive is selected: drive 0 1 2 3 4  
2 4 8 16 128

This program is numbered so that it can be merged between lines 0 and 10. For a stand-alone program you may add the following two lines:

90 RANDOMIZE USR 100: NEW  
9000 RANDOMIZE USR 100: SAVE "booter.B1" LINE 1

1 RESTORE 4  
2 FOR a=16100 TO 16164  
3 READ b: RANDOMIZE USR 100:  
POKE a,b: NEXT a  
4 DATA 128,205,98,0,62  
5 DATA 128,50,3,32,0,0,0,0,0,  
33,27,63,17,34,32,1,10,0,237,176  
62,11,50,2,32,205,198,0,42,124,  
32,34,51,32,42,134,32,34,49,32,2  
05,201,0,62,100,251,201,0,0,0  
6 DATA 65,85,84,79,83,84,65,8  
2,84,32  
7 RANDOMIZE USR 100: POKE 821  
4,16100  
8 BEEP .5,.5: PRINT "NMI-F RE  
BOOT ACTIVATED": PAUSE 100:CLS

How it works:

161XX		
00 243	DI	:disable intrpt
01 205,98,0	CALL 98	:turn on cartridge
04 62,128	LD A, 128	:select drive 4
06 50,3,32	LD (8195),A	
09 0,0,0,0,0		:delay
14 33,27,63	LD HL,16155	:transfer filename
17 17,34,32	LD DE,8226	:to progmn
20 1,10,0	LD BD,10	
23 237,176	LDIR	
25 62,11	LD A,11	:set NC load flag
27 50,02,32	LD (8194),A	
30 205,198,0	CALL 198	:load name
33 42,124,32	LD HL,(8316)	:destin-start addr
36 34,51,32	LD (8243),HL	:store in temp4
39 42,134,32	LD HL,(8326)	:Totlen-length
42 34,49,32	LD (8241),HL	:store in temp2
45 205,201,0	CALL 201	:second data
48 62,100	LD A,100	:exit cartridge
50 251	EI	:enable intrpt
51 201	RET	:return
52 0,0,0		:not used
55-64	characters	:AUTOSTART

Les Cottrell

Cocoa, FL USA

## SPECTRUM MOTHERBOARD

At some time or another, most Spectrum owners realise the need to connect several devices simultaneously to the Spectrum's expansion bus. The normal method which is employed with commercial interfaces is simply that of running the bus through the interface so that the Spectrum's expansion edge connector is duplicated to facilitate the connection of other peripherals which require access to the bus signals.

This seems to work reasonably well when only one or two interfaces are stacked together, however, it can be somewhat problematic when a large number of external boards and devices are present or when external modules do not possess a "through bus" facility. The obvious answer

to this problem is the use of a "motherboard" which makes connection to the Spectrum's edge connector and provides a set of identically wired connectors for external cards and modules.

**Norman Belham** (from Badsey near Evesham) has provided a simple but elegant solution to the problem of constructing a motherboard based on commonly available copper stripboard. Norman writes:

Two pieces of Veroboard are placed back to back (so that the track sizes are exposed) and held together by a Veropin placed in each corner. A convenient available size is 36-strips each with 50-holes (127mm×95mm).

These pieces are large enough to take a Spectrum edge connector along one side, across the copper tracks, and two or three other connectors standing on the surface. If more connectors are required, Veroboard can be obtained 4in. wide and up to 19in. long cut to length from J. R. Hartley of Bridgnorth.

According to "Murphy's Law", anything that can go wrong will and so extreme care is needed in construction! The double Veroboard (arranged so that the copper tracks are outermost) should be inserted between the rows of pins of the edge connector leaving a space, equal to about half a pin length, between the edge of the board and the body of the connector.

The pins should be soldered to the corresponding tracks on the top and underneath the board. It is best to solder pins at each end first so that the connector is correctly located.

When it has been decided just where on the surface of the board the other connec-

tors are to be placed, it is wise to check that they are on the top side when the edge connector is mated with the Spectrum. The pins to be soldered to the top side of the board should be bent at right angles to their mid-point (see Fig. 1 for details). This will make soldering easier when several connectors are in position. A piece of 0.25in. square section beading can be used as a bending bar.

The holes through which the other row of pins reach the lower tracks will require careful counter-sinking using a small drill (1/16in. or smaller). Very light pressure and very few turns are all that is required to produce an insulating "collar" around each hole.

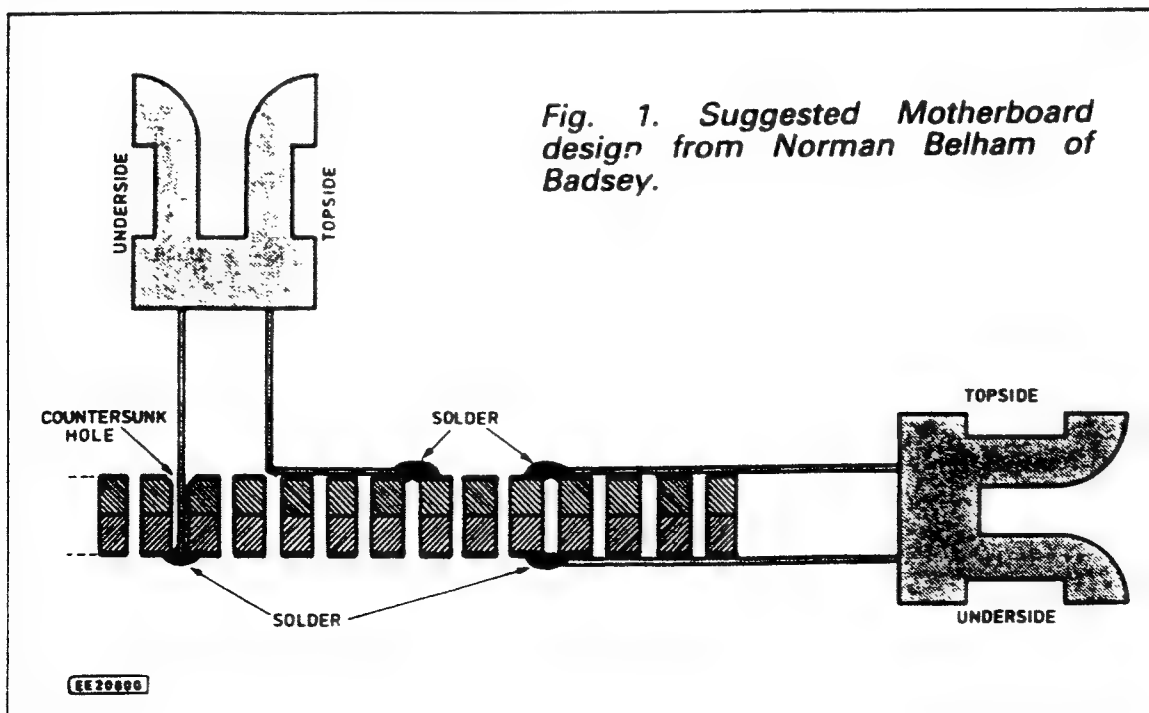
Although the specified drill is too small to cut away the track completely, it is better not to drill the hole completely. (If a track is cut accidentally, all is not lost as a small insulated "jumper" link can be soldered in place). The pins may then be soldered to the lower tracks.

Since only 28 of the tracks are used, the remaining few on either side may be used for other purposes such as an external power supply. Rigorous testing with an Ohmmeter (multimeter set to the "Ohms" range) is essential to ensure that there is no contact between tracks. If, in spite of this, an unusual graphic display is produced when the board is attached to the Spectrum, there is contact somewhere!

It is also wise to test for continuity between the pins which *should* be connected. With such a large number of soldered connections, a "dry" joint may slip by! Finally, since the board does not rest on the bench or table surface when mated with the Spectrum, suitable rubber or plastic feet should be fitted.

The above article also works on TS1000s and 2068s.

This article appeared in the July 1989 issue of EVERYDAY ELECTRONICS, a UK magazine available here. It publishes a Spectrum column every month.



# EOB'S Notebook

**T**erminology requires definitions so that readers may understand what the author is trying to say. In various computing texts, articles, newsletters, even Computer Shopper, often the writers assume that the reader knows all the terms and jargon. Here are a few to help you to find your way through this rapidly growing jungle.

## BANK-SWITCHING.

This is the ability to use more than one set of memory chips at different times, while giving them different addresses. This makes it possible to fit a computer with more memory than it was designed for; simply store some data in one set of chips, then switch over and use the other set. The TS2068 has only 38652 bytes FREE on start up and that's for program and data. By using static RAM chips mounted in the Larken RAMDISK board, it is possible to add up to 256 kilobytes of extra memory for data. Even the IBM PC uses this technique to expand its memory from the basic 640K. Bank switching is also used with the PC on advanced colour graphics cards.

## CHIPS (Integrated Circuits).

These are electronic devices comprising many miniature transistors and other circuit elements on a single silicon wafer about 1/4 inch square. A static RAM chip used on the RAMDISK board will store 32K of data; eight of these will store 256K. The ultimate integrated circuit is the microprocessor which is a single chip that contains the complete arithmetic and logic unit (central processing unit) of a computer.

## RS-232 (Recommended Standard 232C)

defines a standard way of transmitting serial data by wire. A standard RS-232 cable can link two pieces of equipment only if one of them is DCE (Data Communications Equipment) and the other is DTE (Data Terminal Equipment). Most computer terminals are DTE; most modems are DCE. DCE uses conductor 2 for input and 3 for output; DTE uses conductor 3 for input and 2 for output.

## DESKTOP.

The opening screen of a window-oriented operating system is called a desktop because it is a blank space on which various objects can be placed, hence Desktop Publishing, which is being used to do this column.

## BUS.

The bus is the main avenue of communications in a computer. It comprises a set of parallel wires to which the CPU, the memory and all the input-output devices are connected. The bus contains one wire for each bit needed to give the address of a device or a location in memory plus one wire for each bit of data to be transmitted in a single step (usually 8 bits), and additional wires that indicate what operation is being performed. The bus sends data in either direction between any two components. Without a bus, separate wires would be needed for all possible connections between components.

## COMPUTER GENERATIONS.

Computers can be divided into five generations:  
First: built in the late 1940's and early 1950's using vacuum tubes.  
Second: built in the 1950's and 1960's using transistors.  
Third: built using chips; those with large scale chips (100 or more gates) are often called fourth-generation computers.  
Fourth: often viewed as advanced third-generation.  
Fifth: Not yet ready but predicted by some for the 1990's, these will process knowledge instead of just data, eg, natural languages such as English and Russian rather than restricted to programming languages such as BASIC and assembly languages.

## HARD DISK.

A storage medium made of rigid aluminum coated with iron oxide. Now sized up to 40-megabytes or higher, the read-write head travels across the disk on a thin cushion of air without ever touching the disk. **WARNING!** every hard disk in use today WILL break down within just a few years. So make floppy disk backups.

## ISDN.

An ISDN (Integrated Services Digital Network) telephone line is an all-digital line offering the ability to send and receive digital and voice data without a MODEM.

## RECORD.

A RECORD is a collection of related data items. A collection of records is a FILE. A record can be broken down into FIELDS with one each for, say: name, street, city, province, phone number, etc. I include this item because some writers get confused and refer to records as files.

I am indebted to the Barron's Dictionary of Computer Terms Second Edition 1989 for much of the above material. Bob Mitchell. 900113.



Miracle Systems has had a MIDI (Musical Instrument Digital Interface) on the market for quite some time now. This month we'll take a look at a QL MIDI by Miracle Systems.

MIDI interfaces allow instruments such as keyboards, drum machines, etc, to be linked together via a set of cables to communicate with each other. When an instrument is hooked up to a computer it can open a whole new world of possibilities. For instance, you play a piece of music on your keyboard and the computer 'records' the notes and other characteristics of your piece. Afterwards, you can replay the piece in any speed, tempo, or sound available on the connected keyboard. But it doesn't stop there. You can request that the computer play the piece while you play another part, such as harmony. The computer will merge the two parts together when playing back the piece. However, the data is still separate and can be manipulated in any way. This is called 'multiple tracking'. Some computers limit the number of tracks allowed, but QL MIDI lets you lay track over track until you run out of memory. With a decent synthesizer you can have a whole orchestra at your finger tips. Of course there are many more functions available, but I'll concentrate on the ones offered in the QL MIDI package by Miracle Systems.

QL MIDI requires added memory, and a floppy is highly recommended. I'm not sure how much extra memory you ABSOLUTELY need, but I think 256K should be enough to let the program run. The more you have, the better though! I ordered the QL MIDI from Sharps late last spring. It arrived in about 3 1/2 weeks with everything I needed except the cables to link the instrument. At that time it costed \$139.95 US, and should be about the same now (although not advertised in their catalogue, I think Sharps still has some left). I bought some cables from the Shack and was on my way. With no previous exposure to MIDI, I didn't know what to expect, so I just loaded the software in anticipation. The boot program loads an intro screen, checks for the presence of the interface, and executes some loops with a few pokes here and there to fine tune the interface (set timing at I/O to MIDI standard). The exec part is then loaded along with a runtime version of Turbo Toolkit. The program works very much like an 'ICE' environment - you use a joystick in CTRL1 or cursor keys with space to move a highlight bar on the desired option, then press fire/space. It is VERY easy to use. The manual is supplied on MDV and explains the functions quite well.

Performance: The interface scores big here. Ability to record, playback, speed change, time change, filter data, metronome on/off, quantise, full channel support for 16 instruments, full editing of raw data, time & sounds, and other technical goodies. One sub-menu has option to save, load, transmit, and receive data. The transmit/receive part lets you upload/download directly to the instruments memory. Load/Save are options to save whole

musical pieces to a device. When using I/O, you are supplied with the option of MDV or FLP (1 or 2) and OTHER. The 'other' lets you save/load to any other device that has a valid device driver loaded - this means SER, PAR, FRT and so on. This package has everything anybody short of an expert can want. The only thing that I think could be added to give it a final touch would be voice/sound editing. I've seen some packages that will take a sound from the keyboard and display the wave form on screen and allow full editing and manipulation of it. These packages, however, were for the Atari & Amiga computers and were sold as stand alone programs. Maybe it would be asking too much for something like that. Sure, MIDI on the AMIGA and ATARI might have some outstanding features, but QL MIDI does more than enough to satisfy most music lovers (lets not mention the IBM MIDI. For the price of a cheap interface you can get a dedicated QL with QL MIDI and network it to your QL that has a floppy). All in all, a very good product! I would like to hear from anybody that happens to have the QL MIDI or wants to know more about it.

Here's a quick tip that anyone with more than one QL will find VERY useful. You can link up your computers via SER in the way described in the user's manual. When you have them linked, what do you do? Sure you can transfer programs and data, but is that all? NO! You can almost totally control each computer remotely by either of the following methods.

QL1: type BAUD 9200	QL2: BAUD 9200
LOAD ser1	type any command you want executed on the other computer. QL1 will do it as soon as you hit ENTER.
(1st line of each is not really needed)	
QL1: type BAUD 9200	QL2: BAUD 9200
DO ser1	do same as above.

The second method is one that gives a use for TOOLKIT's DO command. The net result is the same. However, you can issue the DO command within a program, but you cannot issue the LOAD command without losing the program that's in memory at the time. The way the LOAD version works is similar to some of the no-line-number programs that are used as loaders for other machine code. If there is no line number, the QL simply executes it immediately. In fact, if you put line numbers ahead of your commands, you will be typing in a program from one computer to another - line by line!

P.S. Did you know that SDUMP can do colour screen dumps to Epson color printers? Ya! It will do a colour dump on my STAR NX1000 color printer (Epson compat.) in 3 different sizes. Works great for those 3D graphs from PSION EXCHANGE!

SEREN RACK1 70 Bardside Ave. N tel. 416-549-6863  
Hamilton, Ontario  
L8H-4W3

\*\*\*\*\*  
 \* OTHER USES FOR THE T/S 2068 LARKEN RAMDISK \*  
 \*\*\*\*\*  
 LARRY CRAWFORD 357 REYNOLDS RD  
 LONDON ONT CANADA N6K 2P8

#### BACKGROUND

The LARKEN RAMDISK banks are all mapped into the DOCK bank of the 2068 even though the board is plugged into the back expansion slot. Consequently, it is possible to load any of the RAMDISK banks with data for a database or spreadsheet. This means that up to 256K more information can be handled and increases the usefulness of the "old workhorse" dramatically. It is also possible to load a DOCK bank with code which simulates a BASIC cartridge (AROS) and to run that code as a BASIC or m/c program. This is what cartridges do.

#### LARKEN BANK SWITCHING

First of all, the 2068's RAM is organized into 8 chunks of 8K each. Normally, with Display File 1 in use, the bottom two contain ROM and the next one the system variables and ROM routines which are moved to RAM on power-up. These 3 chunks should not be disturbed.

The LARKEN RAMDISK board uses the top 4 chunks (4 to 7) to make up a bank. A bank is made active by switching off the HOME RAM chunks 4 to 7 and turning on one of the RAMDISK banks. The active bank is actually treated as a DOCK bank. Thus the RAMDISK board holds up to 8 DOCK banks which can be selected from the keyboard.

#### HOW?

Which chunks are active in HOME and which active in DOCK is controlled by the Horizontal Select Register (HSR) located in PORT 244 (F4 hex). The individual bits which make up the byte in the port determine in which bank the corresponding chunk will be active. A zero bit means the chunk is in HOME. Since we want to use the top 4 chunks in DOCK, we will load the HSR with 11110000 binary (244 decimal). Thus <OUT 244,240> activates the DOCK bank.

#### WHICH DOCK BANK?

LARKEN uses PORT 7 to select the one to be activated. Actually, any bank less than 8 will work since the address lines are not completely decoded. The second page of the LARKEN RAMDISK notes gives the code numbers corresponding to the banks:

Bank	A	B	C	D	E	F	G	H
Code	7	3	5	1	6	2	4	0

Thus <OUT 7,7> will select bank A. Further, LARKEN uses bit 6 of PORT 7 to control the WRITE-PROTECT circuitry of the RAMDISK board. A zero in bit 6 puts the bank in the READ-only mode. To be able to load data into the bank, we must add 1000000 binary (64 decimal) to the bank code number. Therefore, <OUT 244,240: OUT 7,71> will permit us to put data into DOCK bank A.

#### WHAT CAN BE PUT INTO IT?

Strictly code. I use two of the banks to hold records for a database program. It switches the banks in or out as needed to add records or to sort them.

The code, however, can be in the form of an AROS cartridge and therefore be used to hold a BASIC program.

#### HOW TO SAVE THE RAMDISK BANK

Any data read into the bank would destroy the existing data. To preserve the RAMDISK, enter the following: OUT 244,240: OUT 7,7 :RANDOMIZE USR 100: SAVE "RAMDSK.C1" CODE 32768,32768. The contents of bank A will now be safely on floppy and we can mess around with the bank as much as we want. The DOCK bank, however, is still active and needs to be turned off.

#### HOW TO GET BACK TO NORMAL

The commands <OUT 244,0: OUT 7,0> will do the job. With a zero in the HSR via PORT 244, all bits are zeros and all chunks of RAM will be active in HOME. The zero to PORT 7 will turn return to bank H which has a code of zero. This is the normal condition for RAMDISK.

#### BANK H - A SPECIAL CASE

When the computer is turned on or NEWed, it checks the DOCK to see if an AROS is present. If one is, it is automatically activated. Since bank H is really bank 0, a simulated AROS loaded into this bank will run if the <NEW> key is pressed. If you want a permanent AROS in bank zero, install a switch in the line running from pin 1 of the 74145 to pin 20 of the H RAM chip. I have HOT-Z in the socket. Without the switch, HOT-Z would run every time the computer was booted.

If you wish to put an EPROM in the bank, it would be prudent to disconnect pin 28 of the socket from the RAM power supply buss and connect it to the TTL buss. Otherwise, there would be a heavy drain on the backup batteries.

#### HOW TO RESTORE THE RAMDISK BANK

<OUT 244,240: OUT 7,71: LOAD "RAMDSK.C1"CODE> will

put the RAMDISK bank A back in the RAM chip. Don't forget to <OUT 244,0>

#### SIMULATING AN AROS CARTRIDGE

An AROS cartridge must have 8 "overhead" bytes starting at 32768 (8000 hex) to give the computer the information it needs to run the BASIC program. See page 109 of the Tech Manual.

For our demo we will POKE into bank A the following bytes: 1,2,8,128,15,1,0,0.

The first 1 indicates that this is a BASIC "cartridge".

The 2 signals AROS as opposed to LROS, a Language cartridge.

The 8 and 128 form the beginning address of the BASIC program (low byte first). In this case the BASIC will be at  $8+256*128=32776$  or 8008 hex.

The 15 (00001111 bin) means that chunks 4 to 7 will be active in the DOCK bank. This is exactly opposite to the format for the HSR.

The second 1 denotes an Autostart for the program. The two zeros reserve no bytes for m/c variables. The BASIC program will be loaded as code starting at 32776.

#### CREATING AN AROS

1. Save RAMDISK bank A as described earlier and then key in OUT 244,0
2. Key in 10 CLS: PRINT "OK": STOP  
20 OUT 244,240: OUT 7,71  
30 RESTORE: FOR X=32768 TO 32775:  
READ Y: POKE X,Y: NEXT X  
40 DATA 1,2,8,128,15,1,0,0  
50 OUT 244,0
3. SAVE as "test.B1"
4. SAVE as "test.C1"CODE PEEK 23635+256\*23636,1000  
The peeks simply find the address of the start of the BASIC program. It is moved up 50 bytes if the large printer is active. The 1000 is an arbitrary number large enough to be sure that all of the program is saved.
5. Now <GOTO 20>. This will put the "overhead bytes" for the AROS in place.
6. Key in OUT 244,240:LOAD "test.C1"CODE: OUT 244,0. This will put the BASIC program into the "cartridge".
7. Press <NEW> key. This activates the AROS and Line 10 of the program should run.
8. If you saved a RAMDISK bank earlier, load it back in.

#### RUNNING AN AROS (BASIC IN DOCK)

If DOCK bank n contains an AROS, then <OUT 7,bc: NEW> will RUN it (bc is the LARKEN code for bank n). It's that simple.

#### STORING AN AROS

If the code is to be left in a DOCK bank and maintained with the battery backup, it should not be in a bank that was formatted as a RAMDISK bank. If the bank is in Drive 4, then keep the AROS on floppy and load it in when you need it.

#### RUNNING AROS FROM DOCK AND BASIC IN HOME

Programs can exist in both banks and either one can be RUN. Which one is active depends on the contents of the aros flag at 23750. A zero at this address will allow the HOME bank program to be run. A 128 there activates the DOCK program.

A program in DOCK cannot be listed nor can it be edited. So if you attempt to enter a line and can't, you know that you are in DOCK and must return to HOME.

Steps to run both:

1. Boot the DOCK program with <OUT 7,bc: NEW>. This MUST be done first to let the computer know that an AROS is present.
2. Key in POKE 23750,0 to activate HOME bank.
3. Load in the BASIC program.
4. POKE 23750 with 0 or 128 depending on the bank wanted and then RUN or GOTO the appropriate line number.

#### SUMMING UP

The ability to use the RAMDISK banks for other purposes expands the usefulness of the 2068 dramatically. Data that could not be handled by the unexpanded machine can now be dealt with easily.

Huge BASIC programs can be broken down into segments, stored in DOCK banks, and called into action as needed by a short controlling program. This would leave most of HOME RAM free for the storage of data.

Expansion of the 2068 makes economic sense too, removing some of the pressure to move on to a bigger machine that can't be tinkered with.

What we need now is a mess of programs that exploit this potential.

For starters, George Chambers has two of mine. One is a database and the other a HOME/DOCK program handler. Ask him for them.

Explore and enjoy!

Larry C

▶ RANDOMISE USR 100  
▶ 32776



TRIVIAL PURSUIT  
Converting to Larken Disk Operation  
by G. Chambers

Trivial Pursuit is a computer program which is played much like the original board version. In addition to the initial program load, there are 6 further blocks of code which may be loaded, to provide additional game questions. Capturing the initial program to the Larken disk format was relatively easy. Simply a matter of doing an NMI-type SAVE. However, the matter of modifying the program to load the code blocks was more formidable. This article will describe the approach taken.

First, a search was made of the Trivial Pursuit program to locate the LOAD code. The program was broken into by attempting an NMI save to a protected disk. This produced a "disk protected" error report, and we were back in Basic, though with the TR. PURS. code still in the computer. In the possibility that the program used a Spectrum call to the LOAD routine at address 1366, a short for/next program loop was used to look for any sequence of 5 and 86 (5\*256+86) in the program. As luck would have it one was found at address 42932. Then a disassembler program was loaded. I used a TIMACHINE<sub>3</sub> compiled Spectramon program ("spec48.Cc", on club Larken disk #17) loaded at address 48000. On the first attempt the computer locked up. I retried it, this time doing a CLEAR 65000 before loading the disassembler. Seems as though the stack was in the area where the disassembler went. The CLEAR 65000 moved it up out of the way.

I then looked through the code looking for JUMPs to the loading call at 42931. I found several. The LOAD routine is very involved. It is designed to cope with any tape loading errors. If a LOAD error occurs the program will put out a message to that effect, and get back into the LOAD mode again. Hence I found several JUMPs to the loading code. After a number of false starts I settled on address 42647 to insert a CALL to my "disk save" routine.

I should mention that I had written a suitable routine to handle the disk LOAD function. I had also located an empty area in the TR. PURS. code at about address 65360. The starting address for this code would be 65370.

The address 42647 that I mentioned earlier contained a CALL to display a tape load instruction on the screen. Immediately following it was an instruction which when followed led to the CALL to the LOAD routine in the Spectrum ROM. I decided to substitute the existing call (at 42647) with one that would CALL my disk LOAD routine at 65370.

In trying out this change I found that the disk save worked. However it was followed by a tape load attempt. The next thing to do was to bypass this tape load function. In searching the code I found that there were several CALLs in the code which led to address 42931, i.e. the Spectrum ROM loading routine. After a number of false starts I decided to delete the actual CALL to the Spectrum ROM. I inserted three 0's, starting at address 42931. It worked. It seems as though the CALLs to the ROM were bypassed, which was what was hoped for.

Of course I mention that it worked. It was not quite so simple as this. I had to save the blocks of code (the "questions" code) to disk. Although these blocks had a header, the code

refused to load into the computer on its own. Further, none of my header reader programs could pick up the header. Nor could they identify the starting address. All they could identify was the code length. Each block of code was a different length, though they were all about 16700 bytes long.

I located the starting address as follows. First I made a false header by entering a direct command SAVE "test" CODE 30000,16695; and saving the first header part to a separate tape. Then I reloaded this header, moved the TR.PURS. code tape to just past it's header, and loaded the second part of the code. It loaded. I then looked at the first three or four addresses at 30000, and wrote them down.

I mention that I had no knowledge of where the starting address might be. This is not quite correct. In looking at the code with the disassembler I had come across several bits of code which looked as though they might be part of the loading process. The address 43008 was included in several places

The next thing to do, was to confirm that this suspicion was correct. I loaded the program TR. PURS. into the computer from tape, and then loaded the first block of (questions) code. Then I broke into the program and inspected it for the newly loaded code. The first time I tried that I was unsuccessful. The sequence that I had written down was not to be found. Most peculiar. I suspected that the code had been modified in some way or transferred immediately after completion of the load.

On my next try I stopped the computer (using the NMI<sub>3</sub>save/protected disk routine) whilst in the middle of the tape loading process. This time I was successful. I found the code started at the anticipated address 43008.

Now to deal with the code loading routine from disk that I had written. If you look at the disassembled code you can see that first there is a Disable Interrupt instruction at 65370. This prevents the keyboard from interrupting the disk load process. The CALL 98 calls up the Larken DOS, then there is a m/c LDIR routine that transfers the name "genus2.C1" from address 65427 into the Larken DOS at address 8226. After the LDIR routine, the actual load takes place. Following that, at address 65414, there is a routine to increment the number "2" in the title "genus2.C1" (at address 65432) to "3". This is so that the program will load successive code blocks. These code blocks were given the same name, "genusx.C1"; the "x" being "2" for the first block, "3" for the next, and so on with the last block containing a "7". The code loader ends by returning from the Larken DOS at 65421, finishing with the RET at address 65422 which returned program operation back to address 42650 in the TR. PURS.

\*\*\*\*\*

# TRIVIAL PURSUIT Disk Loading Code

65370	F3	DI
65371	00	NOP
65372	00	NOP
65373	00	NOP
65374	CD6200	CALL 98
65377	2193FF	LD HL,65427
65380	112220	LD DE,8226
65383	010A00	LD BC,10
65386	EDB0	LDIR
65388	3E0B	LD A,11
65390	320220	LD (8194),A
65393	CDC600	CALL 198
65396	2100A8	LD HL,43008
65399	223320	LD (8243),HL
65402	213D41	LD HL,16701
65405	223120	LD (8241),HL
65408	CDC900	CALL 201
65411	3A6400	LD A,(100)
65414	3A98FF	LD A,(65432)
65417	3C	INC A
65418	3298FF	LD (65432),A
65421	FB	EI
65422	C9	RET
65423	00	NOP
65424	00	NOP
65425	00	NOP
65426	00	NOP
65427	103	g
65428	101	e
65429	110	n
65430	117	u
65431	115	s
65432	50	2
65433	46	.
65434	67	C
65435	49	1
65436	32	
65437	0	
65438	0	
65439	0	



"Computer still  
in the shop, Jenkins?"

TED TROGDON, SINGER COMMUNICATIONS, INC.

LARKEN PRESENTS ...

UP TO 256K RAM for your 2068

- Expand your 2068 with up to 256K of battery backed up Ram
- Larken Operating system lets you SAVE to memory, just like cassette or disk. (Floppy disk not required)
- All Cassette commands supported. Very Fast and Reliable.
- Can be used with ALL existing 2068 or Spectrum software.
- Uses the new 32K static ram chips, 62256LP or 43256LP
- System consists of Larken Cartridge and Rear Memory Board.
- \*\* PRICE - MEMORY SYSTEM with 64K Ram ..... \$129.00
- MEMORY SYSTEM with 0 K ..... \$ 95.00

## LARKEN 2068 FLOPPY DISK SYSTEM

- The most advanced Dos available for the 2068/Spectrum. LKdos uses ALL Commands such as CAT MERGE ERASE LOAD SAVE PRINT OPEN etc. Also can support RANDISK up to 256K and Sequential / Random Access Files (with additional software). The Larken Disk Interface can handle up to 4 floppys for up to 3.2 MegaBytes of storage. Also NPI Snapshot Save Button and KEYPSTON Joystick port on interface. Also 10 Extended Basic commands for Windows and Graphics.

AERCO RAMEX or OLIGER Disk users can add LKdos for more commands, Randisk and access to all LKdos software

- \*\* PRICE - Larken Floppy Disk System ..... \$119.95
- Floppy Disk IF with 0 K Memory board .. \$169.95
- Larken Disk Editor ..... \$ 15.00
- Sequential/Random access files ..... \$ 15.00
- Xcodes to Disk Modem package ..... \$ 15.00
- ZX81 Floppy Interface (15 left) ... \$ 99.95
- LKDOS for Aerco,Ramex or Oliger Disk IF \$ 59.95

(All prices are US, Add 6% Shipping)

LARKEN ELECTRONICS RR#2 NAWAN ONTARIO CANADA K4B-1H9  
(613)-835-2680

# SHARP'S IS THE LARGEST QL SOFTWARE AND HARDWARE DEALER!

WRITE FOR FREE CATALOG

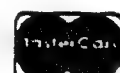
Sharp's, Inc.



Rt. 10, Box 459

Mechanicsville, VA 23111

(804) 746-1664 or 730-9697



Kris Boisvert  
Eric Boisvert  
COPYRIGHT  
BYTE POWER

A TRUE FAST LOAD OPERATING  
SYSTEM! CHECK OUT THESE  
FEATURES...

- ONLY 1536 BYTES LONG AND RELOCATABLE ANYWHERE BETWEEN 31800 AND 64000!
- TS2068 AND SPECTRUM 48K COMPATIBLE!
- HANDLES ALL STANDARD I/O COMMANDS.
- BORDER FLICKER COLORS SELECTABLE. (ANY COLOR AT ALL!)
- NEW COMMANDS SUCH AS NO MESSAGE AND HEADER READER!
- BAUD RATES FROM 1500 UP TO 3300
- USES A PRINT #4: I/O COMMAND

2+FAST.....\$22.95 ppd

UNLEASH THE POWER OF YOUR  
PRINTER WITH THIS SOFTWARE  
PACKAGE...

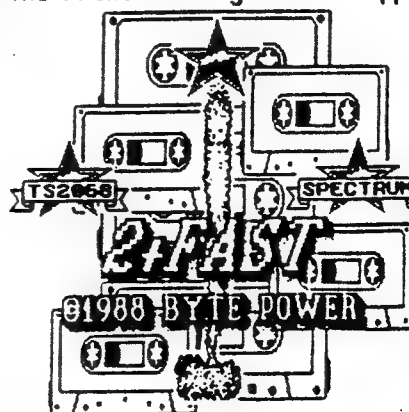
8 PROGRAMS AND OVER 150  
GRAPHIC DESIGNS!

THE LABELER, THE CARD MAKER,  
THE BANNER PRINTER, THE PRESS,  
THE CALENDAR, THE HEADER,  
QUICK SCREEN AND THE CREATOR  
ARE JUST WHAT YOU NEED TO MAKE  
YOUR PRINTER COME TO LIFE!

COMES ON 2 TAPES IN A HIGH  
QUALITY VINYL ALBUM. INCLUDES  
SUPERB INSTRUCTION MANUAL.

VERY USER-FRIENDLY!

The Print Factory...\$22.95 ppd



Also available from BYTE POWER

FIRST CLASS FONTS II...\$22.95  
THE D.U.S. V2.0 (LKDOS)...\$22.95  
THE VOICE...\$22.95  
THE PRINT FACTORY...\$22.95  
2+FAST (FAST LOAD)...\$22.95  
ZX81 1ST CLASS PROGRAMS...\$12.95

BYTE POWER MAGAZINE...

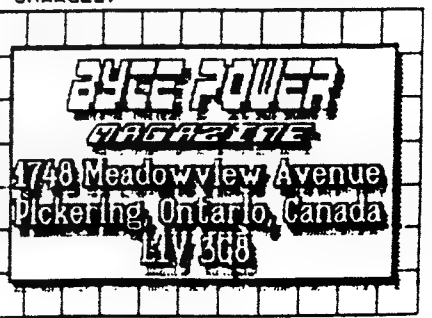
1 ISSUE (BACK/CURRENT)...\$5.95  
6 ISSUES...\$35.00  
12 ISSUES...\$55.00

FIRST CLASS PACK.....\$40.00

BEST OF ARCADE GAMES...\$22.95  
BEST OF BUSINESS...\$22.95  
BEST OF BOARD GAMES...\$22.95

DEMO TAPE/CATALOG.....\$3.00

U.S. FUNDS ONLY. ALL PRICES  
INCLUDE SHIPPING/HANDLING  
CHARGES.



## QLIPS

The other day I had a problem and strange things were happening. Lockups, crashes, some corruption. I eventually changed my Disk Drive #1 to #2, and #2 to #1, as my problem appeared to be when I used a certain disk. In the process I had to open them up to make the necessary changes, and while doing this I noticed a couple of little plugs appeared to be sliding off the sockets, tightened them up, and this cured my lockup lost-data problem.

Next day, my problem was somewhat different, in the middle of a programme, there would be blue flashes crossing the screen. My first thought was how the solar flare activity could do this to me, so I scrubbed this idea, and thinking I had perhaps altered something on my Disk change-over, went through all the cable connections I could think of. Whatever happened the flashing went away.

Then we had another problem. On power up, the screen on my RGB would go GREEN, and that was it. At this point I resorted to my old radio days when we had all those little bottles (called TUBES or VALVES) inside a receiver or transmitter. I gave the table a thump, and the monitor a bump, and lines disappeared.

Later I just happened to come across an article in an old CURRY report, and I thought it might be of interest to someone out there. The issue was Vol. 2 No 11 Dated November 15.1986. and I would like to pass this on.

Here is the QOUTE from page 5 :-

"The most common 'downtime problem' with QL's seems to be either bad microdrives or video problems. Video is usually confined to faulty ram chips or a defective ZX8301. With early or 'low' serial numbers, the ZX8301 chip could be 'popped out' very easily from its socket. Unfortunately with higher serial number QL's, a lot of 'superfluous' material in the way of wires, etc. cover the top of the chip. Needless to say, this makes it much harder to replace. Generally speaking, if on power up you get a "green screen" it is a problem with ram chips. If you power up, get a logo screen, etc. and then the video goes down (slowly or otherwise), it is probably the ZX8301. [ see Service Manual]

For those of you who suspect the 'lockups' your QL experience is due to the voltage regulator, you might try removing the 7805 and replacing it with a 78S05 which is bigger and stronger. This was mentioned by a QUANTA member in Kenya. You can also add... a 0.1mf capacitor. This sequence does not even need the use of a soldering iron. If you open the Keyboard, the 7805 will be screwed onto the heatsink which is located behind the microdrives. Undo the heatsink screws, unplug the 7805 from the flying leads, and remove the unit. Then, screw a new regulator on and while you are plugging the leads back in, you can put the wires from the capacitor in that same socket. "

End of quote from Curry report vol. 2 no 11.

So there you are, was my thump on the table enough to re-settle the chips to where they should be ? I don't know, but at least the blue flashes have not come back, and my screen does not go green now.

By the way, we have a number of the CURRY REPORT on hand, if you are interested in reading them let me know. Lots of goodies.

On the subject of QL problems, ( Is there any other QL subject?) Say you have a Miracle 512 expansion, and then a Cumana interface plugged into this, you would be well advised to use a support at the extreme left end of the ADD-ONS, an old microdrive case is just about the right size. It must be remembered that the connections are very short and give no support whatever to the extensions, hence the need for a little support.

Another idea is to ensure your QL LEGS are long enough. After a period of use, feel the underside of the QL, and you will find it is quite hot, mine is, and I have a good space between the QL and the table. I Can't imagine the build-up of heat if it were not for the longer legs.

H. H. H.



## VIDEOTEX

- > Written for use with the S.M.U.G. Video Digitizer.
- > Capture and display thirteen different full-screen greyscale shadings of a digitized video image.
- > Greyscaled images can be inverted, horiz. flipped, filtered, and saved for use with drawing programs.
- > Special "overlay" option allows for even more creative possibilities and special effects.
- > Complete User Notes plus information relating to customizing VIDEOTEX for other storage media and full-sized printers is included.



Above greyscales  
made w/VIDEOTEX.

## VIDEO 3-D

- > Generates 3-D representations of digitized video images saved as master video files with VIDEOTEX.
- > Normal and "inverted" 3-D effects are possible.
- > Four selectable Z-axis "depths".
- > A 3-D representation can be auto-colored with either of two color palettes.
- > A 3-D representation can be inverted, horizontally flipped, and saved for use with drawing programs.
- > Complete User Notes plus information relating to customizing VIDEO 3-D for use with other storage media and full-sized printers is included.



normal 3-D

Made with VIDEO 3-D

inverted 3-D



## VIDEOCOPY

- > For hardcopying digitized video images saved as master video files with VIDEOTEX to an OKIMATE 20 printer (using an IBM parallel Plug 'n Print cart.). Not for use with other color printers or other OKIMATE 20 Plug 'n Print cartridges.
- > Both black & white and hi-res auto-colored hardcopies are supported.
- > 13 different full-screen greyscalings can be viewed, inverted, and horizontally flipped before hardcopying.
- > Automatically colorized digitized video images can be copied.
- > A "Color Editor" is available for user-assigned "colorizing".
- > Complete User Notes plus information relating to customizing VIDEOCOPY for use with other storage media is included.

NAME: \_\_\_\_\_

ADDRESS: \_\_\_\_\_

CITY/STATE/ZIP: \_\_\_\_\_

QUANTITY	SOFTWARE TITLE	PRICE EACH	TOTAL
	VIDEOTEX	\$9.95 ppd	
	VIDEO 3-D	\$9.95 ppd	
	VIDEOCOPY	\$9.95 ppd	

(Pre-paid orders only please.) TOTAL ENCLOSED>>>

SEND ALL ORDERS/INQUIRIES TO:

John McMichael, 1710 Palmer Dr., Laramie, WY 82070

# S.M.U.G. Presents

The 1990 **SINCLAIR COMPUTER** Exposition

## MILWAUKEE

## WISCONSIN

June 2 & 3 / Banquet Friday Night June 1  
**SEMINARS, DOOR PRIZES, SWAP SHOP,  
SOFTWARE, PERIPHERALS, HARDWARE,  
AND LOTS OF OTHER STUFF**

Location

WAUKESHA HOLIDAY INN (414) 786-0460

Hwy 18 & I94 Waukesha, WI 53186

There will be a **SNUG** meeting Saturday Night June 2, 1990

Ticket Information:

	<u>in advance</u>	<u>at the door</u>	
One day	: \$ 4.00	\$ 5.00	
Both days	: \$ 7.00	\$ 9.00	
Banquet	: \$16.00	\$16.00	Limited seating
Table	: \$25.00 (incl. 2 day badge)	\$25.00	Limited table space

Tables are 6 feet by 30 inches

For **MORE INFORMATION** contact

Bill Heberlein

Neal Schultz

5052 N. 91st Street

or

call 7 - 10pm

Milwaukee, WI 53225

(414) 353-4522

for **RESERVATIONS** mail to:

Expo Reservations

P.O. Box 101

Butler, WI 53007



Detach and Mail

Name \_\_\_\_\_ Phone (\_\_\_\_) \_\_\_\_\_

Address \_\_\_\_\_

City \_\_\_\_\_ State \_\_\_\_\_ Zip \_\_\_\_\_

Please reserve \_\_\_\_\_ one day badge @ \$ 4.00 each = \$ \_\_\_\_\_

Please reserve \_\_\_\_\_ two day badges @ \$ 7.00 each = \$ \_\_\_\_\_

Please reserve \_\_\_\_\_ Banquet tickets @ \$16.00 each = \$ \_\_\_\_\_

Please reserve \_\_\_\_\_ 6' x 30" tables @ \$25.00 each = \$ \_\_\_\_\_

I have included a check/money order for total = \$ \_\_\_\_\_

## TORONTO TIMEX-SINCLAIR USERS CLUB

March 17, 1990

14 Ridgeway Court,  
Scarborough, Ont. M1R 2Y1Les Cottrell  
108 River Heights Drive  
Cocoa, FL 32922

Dear Les:

Sorry I have not got back to you sooner. It's been so long that I can't find your letter to reply to it.

Not that you have been lost or forgotten! You will see that your newsletter article has been published. Well, actually I see where the editor has published two of them. I guess he does not like to save things for a rainy day or something. I see where two of my articles are in the same issue, and I wish he had kept one of them for the next month. I can't just whip up one of them out of thin air. Can you?

Well, I sent your article to Bob Mitchell shortly after I received it from you. He incorporated it into his OMNIBUS disk, so that when loading the AUTOSTART menu it booted the routine into the DOS. However, Bob went a bit further. He buried the code in the AUTOSTART program at address 24495. That way he does not have to hold it in the BASIC part of the menu program. Just do a POKE routine from BASIC to move it from 24495 to 16100.

I carried the idea a bit further by putting an LDIR routine in front of it at the address 24495. Then in the menu program I just make a USR call to 24495 and it gets booted into the DOS. This whole exercise does two things. It gets the routine into the DOS quickly, and it uses empty space in the AUTOSTART program (24495 & upwards of 80 bytes) rather than in the BASIC program itself. The AUTOSTART program has to have a line to do the USR call to 24495, and also a POKE into the DOS to install the starting address 16100.

Because Bob has some other m/c routines that he POKES into the DOS he moved the routine to address 16310, but that is just a minor point. That sort of consideration resulted in choosing 24495 as the appropriate storage area for the routine. I'm sending a listing so that you can try it out. Do you want to write up an article about this development, or shall I. I see where, if one chose one could point the operation to any drive you chose. That might be useful to members who do not have a RAMdisk.

If, or I should say, when I come across your letter I shall respond to it. Are you waiting for something from me? I'll cast around before I send off the n/1 to see.

Sincerely,

George Chambers.

We do appreciate the article. You see how it can prompt a whole new set of activities. That's part of the problem; I've been playing around with it instead of getting at my correspondence!



2 REM by Les Cottrell,  
with mods by Geo Chambers

15 RESTORE 30: FOR a=24495 TO  
24575

20 READ b: POKE a,b: NEXT a

30 DATA 243,205,98,0

40 DATA 33,195,95,17,228,62,1,  
58,0,237,176,58,100,0,251,201

50 DATA 243,205,98,0,62

60 DATA 128,50,3,32,33,20,63,1  
7,34,32,1,10,0,237,176,62,11,50,  
2,32,205,198,0,42,124,32,34,51,3  
2,42,134,32,34,49,32,205,201,0,6  
2,100,251,201,0

70 DATA 65,85,84,79,83,84,65,8  
2,84,32,0,0,0

80 RANDOMIZE USR 24495

90 RANDOMIZE USR 100: POKE 821  
4,16100

100 BEEP .5,.5: PRINT "NMI-F Re  
boot activated": PAUSE 0: CLS  
8999 STOP

9000 RANDOMIZE USR 100: SAVE "NM  
I-F.B3"

9010 REM This SAVE puts the code  
into DOS at address 16100

24495 F3	DI
24496 CD6200	CALL 98
24499 21C35F	LD HL,24515
24502 11E43E	LD DE,16100
24505 013A00	LD BC,58
24508 EDB0	LDIR
24510 3A6400	LD A,(100)
24513 FB	EI
24514 C9	RET
24515 F3	DI
24516 CD6200	CALL 98
24519 3E80	LD A,128
24521 320320	LD (8195),A
24524 21143F	LD HL,16148
24527 112220	LD DE,8226
24530 010A00	LD BC,10
24533 EDB0	LDIR
24535 3E0B	LD A,11
24537 320220	LD (8194),A
24540 CDC600	CALL 198
24543 2A7C20	LD HL,(8316)
24546 223320	LD (8243),HL
24549 2A8620	LD HL,(8326)
24552 223120	LD (8241),HL
24555 CDC900	CALL 201
24558 3E64	LD A,100
24560 FB	EI
24561 C9	RET
24562 00	NOP
24563 41	LD B,C A
24564 55	LD D,L U
24565 54	LD D,H T
24566 4F	LD C,A O
24567 53	LD D,E S
24568 54	LD D,H +
24569 41	LD B,C A
24570 52	LD D,D R
24571 54	LD D,H T
24572 2000	JR NZ,24574
24574 00	NOP

```

32>PRINT #od:GO TO ob:GO SUB V
AL "11":RANDOMIZE USR VAL "24495"
":PRINT #od:POKE VAL "8214",VAL
"16310"
33 LET ol=of+of: LET
0": LET

```

*N/A - F*

16310

```

24495 F3 DI
24496 CD6200 CALL 98
24499 21C35F LD HL,24515
24502 11B63F LD DE,16310
24505 014100 LD BC,65
24508 EDB0 LDIR
24510 3A6400 LD A,(100)
24513 FB EI
24514 C9 RET
24515 F3 DI
24516 CD6200 CALL 98
24519 3E80 LD A,128
24521 320320 LD (8195),A
24524 00 NOP
24525 00 NOP
24526 00 NOP
24527 00 NOP
24528 00 NOP
24529 21ED3F LD HL,16365
24532 112220 LD DE,8226
24535 010A00 LD BC,10
24538 EDB0 LDIR
24540 3E0B LD A,11
24542 320220 LD (8194),A
24545 CDC600 CALL 198
24548 2A7C20 LD HL,(8316)
24551 223320 LD (8243),HL
24554 2A8620 LD HL,(8326)
24557 223120 LD (8241),HL
24560 CDC900 CALL 201
24563 3E64 LD A,100
24565 FB EI
24566 C9 RET
24567 00 NOP
24568 00 NOP
24569 00 NOP

```

LOAD PGM  
NAME  
"AUTOSTART"

```

24495 243
24496 205
24497 98
24498 0
24499 33
24500 195
24501 95
24502 17
24503 182
24504 63
24505 1
24506 65
24507 0
24508 237
24509 176
24510 58
24511 100
24512 0
24513 251
24514 201
24515 243
24516 205
24517 98
24518 0
24519 62
24520 128
24521 50
24522 3
24523 32
24524 0
24525 0
24526 0
24527 0
24528 0
24529 33
24530 237
24531 63
24532 17
24533 34
24534 32
24535 1
24536 10

```

3 lines affected.

# MY OMNIBUS MENUS

new: ① → 10: GO TO VAL "30"  
11: PRINT #od: POKE oz, VAL "16310": LET b=USR oy: RETURN  
12: BEEP VAL ".005", oj+oj: RETURN  
13: IF IN VAL "127"=VAL "237" THEN LET in=oa: RETURN

alter: ② → 32: PRINT #od: GO TO ob: GO SUB VAL "11": IF b=VAL "123" THEN FOR  
i=VAL "24495" TO VAL "24560": PRINT #od: POKE i-VAL "8185", PEEK i:N  
EXT i: PRINT #od: POKE VAL "8214", VAL "16310"  
33: LET ol=of+of: LET ft=VAL "40": LET fh=VAL "500": LET of=VAL  
"1000": LET new=VAL "490": LET pr=VAL "296"  
34: GO SUB VAL "982"  
35: LET lpd=oa: LET drv=ob  
40: BORDER og: INK og: PAPER og: CLS

new: ③ → 41: PRINT #od: POKE VAL "8214", VAL "16310"  
45: PRINT #od: GO TO od: LET l\$="rd1.c1": PRINT #od: LOAD l\$COD  
E  
50: DIM d\$(ol)  
65: IF lpd THEN PRINT #od: OPEN #3, "lp"

\* ① 205, 98, 0, 62, 128, 50, 3  
32, 0, 0, 0, 0, 33, 237  
63, 17, 34, 32, 1, 10, 0, 237  
176, 62, 11, 50, 2, 32, 205, 198  
0, 42, 124, 32, 34, 51, 32, 42  
134, 32, 34, 49, 32, 205, 201, 0  
62, 100, 251, 201, 0, 0, 0, 65  
85, 84, 79, 83, 84, 65, 82, 84  
32, 0, 0, 0, 0, 0, 0, 0

vice 27.

## code disassembly

16310	56100	7B	LD	A, E	
	56101	CD6200	CALL	98	
	56104	3E80	LD	A, 128	
	56106	320320	LD	(8195), A	
	56109	00	NOP		
	56110	00	NOP		
	56111	00	NOP		
	56112	00	NOP		
	56113	00	NOP		
16325/6	56114	21123E	LD	HL, 16155	← 1631
	56117	111220	LD	DE, 8226	(23), 6
	56120	010A00	LD	BC, 10	
	56123	EDB0	LDIR		
	56125	3E0B	LD	A, 11	
	56127	320220	LD	(8194), A	
	56130	CD6600	CALL	198	
	56133	2A7C20	LD	HL, (8316)	
	56136	223320	LD	(8243), HL	
	56139	2A8620	LD	HL, (8326)	
	56142	223120	LD	(8241), HL	
	56145	CD6900	CALL	201	
	56148	3E64	LD	A, 100	
	56150	FB	EI		
	56151	C9	RET		
	56152	00	NOP		
	56153	00	NOP		
	56154	00	NOP		
16365	56155	41	LD	B, C	
	56156	55	LD	D, H	
	56157	54	LD	D, C	
	56158	4F	LD	C, A	
	56159	53	LD	D, E	
	56160	54	LD	D, H	
	56161	41	LD	B, C	
	56162	52	LD	D, D	
	56163	54	LD	D, H	
	56164	20			

- ① GET value in LKDOS 16310.
- ② code poked into LKDOS RAM @ 16310 from 2068 RAM @ 24495 where it was stored from my copy @ 56100; but only poked if b < 123. \* So initial menu delay only occurs once.
- ③ LKDOS addr. 8214 restored with 16310; it may have changed (eg. my GRAFLX.B1 makes it 16100).

R.A. MITCHELL

Jan/Feb 1990

January 11th, 1990  
cont'd January 27th, 1990

Dear Out-of-town members,

I'm writing this before the newsletter has come out, so I can't comment on it directly. But from the look of the material that members have been sending in, it's likely to be pretty good. We do appreciate the work of those members who have submitted material. I would like to call on those who have not contributed yet, to please do so. It helps immensely.

The reason our newsletter is late has a bit to do with Christmas time, and a lot to do with an event in our Editor's life. Through the efforts of his wife he is now the proud father of a baby boy! Arrived just after Christmas. So you can imagine that he has been busy. Jeff now has a girl and a boy in his family.

His address is 335<sup>75</sup> Lemonwood Drive, Islington (Toronto) M9A 4L3 CANADA. If you are inclined, drop his wife a note; she did most of the work!

One of our former members, Ariel Frailich, called me a few weeks ago to tell me that he was interested in clearing out his old Timex stuff. I said that I would inquire amongst members and see what could be sold. I managed to locate enough persons in the local area to take most of the stuff, so one day I went over and picked it up. There was a 2068, a Portuguese Disk system, an Interface 1 with a twister board and microdrive, a 2050 modem, a 2040 printer, a monochrome monitor, numbers of books and magazines, and 7 tape recorders.

I managed to clear most of it out at our January meeting. Good job too, because it was tending to clutter up my basement! I had the two systems up and running, to see how they worked. I still have a few odds and ends, if you wish to inquire.

I mentioned that Ariel was a former member of our club. He, along with Dave Ridge were the two partners in Novelsoft, the outfit known for ARTWORX, ZXPRT, TIMACHINE, THE WORX, and some other programs they wrote and used to sell.

Ariel tells me that he is working on a refined version of ZXPRT for use on MSDOS machines. It has an application in the educational system as a teaching tool. That is, a student works up his study subject into a framework of ZXPRT. The idea being that in doing so, the facts of the subject become fixed in the mind and at the same time the student becomes aware of gaps in his knowledge of the subject he is studying. Quite a number of school boards have been interested enough to buy copies of the program. It comes with text material for the student and the teacher. Rather a neat approach as a teaching aid, I thought.

It is interesting to note that along with Dave Ridge and Ariel Frailich, our club can also include Cameron Hayne, the author of TIMACHINE, as a past member.

I have not added any disks to our Larken library recently. The number stands at 15 disks. But I have some disks in progress that you might like to see. They include disks on the topics of Banners, Menus, Languages, and Calendars. If any of members are interested in seeing a copy, drop me a line.

Also a couple of items about disk #6, Music & Sound. In going over it I found a large number

of flaws in it. I had to go completely through it, to fix up the problems. Mostly they were to do with making the programs work from the directory. For example I had to put a RESTORE function in some programs so the would not come up with an 'OUT OF DATA' report. So if you have borrowed that disk and thrown up your hands, re-borrow it. Also, while on that same disk: It has several music programs by Joan Kealy. I really liked them so I wrote to her, asking if I could get some more of them. She has obliged, and I am putting them all onto a separate Larken library disk. Ask for it; I have not given it a library number yet. In addition to sending me all these music program, she has taken a subscription to our newsletter. Welcome Joan, I hope you enjoy the newsletter!

I also will have a disk with one or two versions of the Interbank Data Storage/Sorting Utility. This is the program I mentioned in my last newsletter; author Larry Crawford. It is a huge database file, which bankswitches the Larken RAMdisk chips into the computer memory as needed to store and access the data. Can hold up to 9000 or so 27-character records. There have been some refinements made to the program since I last mentioned it. Bob Mitchell has refined the front-end, making it more user-friendly; while Larry Crawford has done the same, and also modified it so that you can specify the record length when you start a new file, up to a maximum of 128 characters. Also Larry has modified the SORT routine so that it will sort on any designated column. It has a remarkably fast SORT routine, and a fast string SEARCH routine built-in. Bob plans to use one version to hold a record of his colour slide collection.

I have learned an interesting thing. I have a Smith Corona L<sub>1</sub>1000 printer. I have mentioned it before. It is a daisywheel type printer, very ruggedly built. A week ago when I need a new printwheel for it, I was unable to do so. 'Not available' they said, and not being manufactured any more. I was dumbfounded. I'm told the printer is very old; I bought it new in 1985! The printwheel that I was asking for was a Red Ring Pica 10, the most common type, nothing special. They did not have any of them, in fact they had no RED RING types at all; no printwheels at all for the L<sub>1</sub>1000 printer.

I settled for a White ring Judicial 10 printwheel. It is not made for the L<sub>1</sub>1000, and is not really satisfactory, since the print hammer tends to strike slightly off-centre. Anybody out there own one of these beasts; had any suchlike problems? Moral, don't buy any Smith-Corona product; you'll get short shrift from them. Any ideas, anyone?

You may see a letter in the newsletter from Bill Jones (UPDATE magazine). It responds to a column by Hugh Howie, in the previous letter. I found it so amusing that I felt it deserved a wider audience. There are a couple of interesting things about it.

Firstly, I had to thank Bill for his offer to put a club advert in his magazine, but to say that we would decline for the present. That I had all the members that I could handle just now, and that I did not need any more. We are still getting queries (and new members) from the mention that Bill has put in, in past issues. That's interesting, isn't it! I'm just too afraid of loosing an avalanche.

The second thing was Bill's reference to



'snoring'. Maybe 40 years ago, I had read of this technique, and one rainy day I decided to give it a try. My wife (I was newly married then) looked out the window, and saw me in a raincoat, doing the most odd thing. When I tried to explain just what it was, she couldn't believe what she was hearing. Though I maintained at the time that it was very serious business, and continue to do so, she, to this day, regales all and sundry about that event. And nobody really believes my version; they won't take my protestations seriously.

So you see, when I read Bill's letter, I ran to my wife, calling out, "Listen to this! listen to this!". But she still treats me indulgently, and I don't suppose that she'll ever tire of telling that story (her version, of course!). And I don't even fish.

I see where the COMPUTER SHOPPER magazine has dropped Michael O'Brien's Timex/Sinclair column. When I look further I see an explanation in the Editorial (page 197). They have dropped all the 'classic computers' columns from the magazine, saying, "We have chosen to discontinue our coverage of these machines in favor of.....those systems that more fully represent the mainstream of modern computing." I suppose that is the consequence of being taken over by ZIFF, DAVIS. Shades of SYNC magazine.

Further, I see where the Editor, Stan Veit, has been kicked upstairs. That is to say, he is now called 'Editor in Chief Emeritus'. I call on some of you professionals to tell me what that really means. I would hazard a guess that he wasn't happy with the turn of events, and was given the shove.

In their editorial they continue, to quote, 'We hope they (meaning us) continue to enjoy and use Computer Shopper as they join their computing colleagues and migrate to IBM, Mackintosh, Amiga, Atari ST, Apple IIGs, and Unix systems'. end of quote

Bob Mitchell dropped me a line a few days ago. He reports that Larry Kenny mentions working on a DOS that will read and write to MS-DOS disks, 'just for transferring programs and not for making them run', to quote from the letter. (Myself, I rather suspect that it would be most useful in the transfer of ASCII-type text.)

Larry also make mention of the Desk Top Publisher program he is working on, saying it will be ready in a few months.

In another club's newsletter, I read that Radio Shack were discontinuing the paper for their TP40 printer. That is the same paper we buy for our TS2040 printers. I checked at the local RS store. They had it in stock, and when I inquired, they looked up their data base to see if it was under review. It was not, so we in Canada may be secure for a while yet. I understand from them that the Canadian R.S. operates independantly of the US operation.

Our newsletter carries only one mailing address in it; mine. There's nothing particularly wrong with that but if anyone should wish to complain (about me, for example) you'd be screened pretty closely!! There haven't been any complaints (not recently, anyway!), but I'll give the address of our club President, Rene Bruneau. It is:

120 Salem Ave.,  
Toronto, Ontario,  
CANADA M6H 3C3

I think I am pretty much up in my mailings. If you feel you have been waiting too long, drop me a line; maybe I have overlooked you.

Just a mention of something. If you are returning tapes, please be sure to rewind them completely. If they are returned otherwise, it is quite possible for the tape to get damaged. The next person to get the tape will then have a problem.

Being with the club all this time I have acquired a great many books, clippings, magazines, programs, and just such an array of stuff that it is pretty nearly impossible to let you know what I have. If you want information, just ask me; chances are that I can come up with something for you. That is what a club is all about. I have been fortunate in getting a lot of stuff from members over the years, and I really would like to share it with you.

One of our members sent me some small pieces of paper off the last package that I sent him. I found them very useful. There seems to be quite a number of them on packages sent to the US. Other members might do likewise!!

I have revised my Income tax program for the tax year 1989. It is suitable for the Canadian situation; more specifically it handles the Canadian and Ontario Income tax forms. Ask for a copy if you are interested.

I picked out an advert that I thought particularly humorous. Sort of a black humor, if you will. I'll tack it onto the letter. Maybe because I do volunteer work for the blind, repairing 'Talking Book Machines' (tape players, to you), it struck my fancy!

35, or	TV-unwanted prize, 21" b & w, good working cond., suitable for people with impaired vision. \$50. 482-4442.	colle SELN 232-92 TDIIM
— 95, 99, — cel.	TYPEWRITER	

I have a member's letter that I want to publish in the next n/l, but I'd like to put a query from it right here. I'll paraphrase his query: "I have bought a SAMSUNG MZ4571 14" monochrome monitor. It is a flat screen, dual mode (RGB/TTL) monitor. Since we do not have TTL compatibility with the TS2068, I'm using the RGB mode. There is one minor problem with it... if the border is any color other than black, the top four or five display scan lines roll down across the screen. I have gotten into the case and tried adjusting things but with no success. I wonder if a different RGB i/f would help? Any ideas?". Any one got any ideas, do drop me a line, G.F.C.

One other thing. I seem to be missing two pages from my MSCRIPT manual. They are pages 39 and 40. Maybe these pages were not part of the TS2068 version of MSCRIPT, but the manuscript continuity seems to require them. Can any one help me by sending the two missing sheets?

Sincerely

George Chambers.

can not confirm



# TS Bulletin

News Supplement #1-1990 Jan., 1990  
No.1-1990

--Pub.by B. Harmer, 97 Ruskin, Ottawa, Can. K1Y 4B3

News for this year, 1990, may see Sinclair computers out-doing even previous glories. Rumours are that Apple Macintosh and NeXT computer emulators for the QL are being readied by software developers, to follow the MS DOS and CP/M emulators already available. Cambridge is rumoured to have a 3½ lb., MS DOS laptop with 3½ inch disk drives and optional hard disk. But the big news is that most of us are still here, looking forward to another year of user group activity. Ottawa-Hull TSUG has a BBS now, (613)745-8838 (afterhours), 300 baud, 8/n/1, Xmodem, planned to be upgraded to 1200 baud. The writer is still working on photocopy-published Sinclair books, including additions to the ZX-81 Assembly Language Booklet and Tips, Tricks & Techniques from the User Group Masters for the ZX-81/TS1000. A third book, Master Guide to the LDOS ZX-81 Disk System, is winding up final stage of preparation, altho it looks like it will be expensive to sell the full-length, extended edition, at about 100pp., for reasons of its size alone (photocopying & mailing expenses running over \$25). Local group is showing renewed interest in Pascal programming and work on a simplified compiler (NOT BASIC nor Pascal, but a sort of development system macro language) is continuing for the ZX-81/TS1000. The QL is still being advertised at US\$100 (by Sharp's). New ZX-81 kits are still available too from one supplier. The Z-88 notebook computer still looks good in comparison to its latecomer competitors. Commodore 64 and Tandy 1000 H? computers are almost being dumped on the market at bargain basement boxing day time specials (C-64, \$169, Tandy \$399). The computer revolution is seeing a new round of discounting of the cheaper models and an increase in sophistication in the \$1000-\$3000 models. Even Macintoshes are penetrating the \$2000 price barrier in Canada. Faster modems at 2400 bps are being seen in stores, Radio Shack is calling its 80286 MS DOS computer its most popular beginner's model (though it is not AT compatible) and laser printers have reached the small business/serious amateur price ranges. In the big leagues, the slowness of OS/2 to reach its intended place in the market and the IBM PS/2 computers to become the new standard, once touted as the success stories of the foreseeable future has again proven that the future isn't foreseeable at all. This year is also the tenth anniversary of the release of the Sinclair ZX-80 and the start of the Sinclair saga in computing.

Mailing list updates: Harrisburg Area TSUG (HATS) may be deleted from our lists since the Dec., 1989 was the final newsletter and notification of dissolution of the group. Former members will operate through the Washington, DC group (CATS). A new listing in the Computer Shopper has alerted us to the existence of Quantum Leap User Group, c/o Gale Henslee, 4411 2nd Ave., Amarillo, TX, USA 79106. The Dallas/Fort Worth users are still going strong altho they don't appear to be still issuing a newsletter to exchange with other groups. They support especially the Aerco TS2068 disk system and the CP/M compatible RP/M operating system.

News from SNUG is that due to the illness of a member's wife and personal situations of others involved in the group, work on the newsletter has been delayed and all memberships have as a consequence been extended 6 months (that is into 1990). Looks like they will soon have a really terrifically large public domain program library for the QL, TS2068 and TS1000, but they are still looking for more donations of programs for it. It looks like a good way to consolidate the libraries of the local user groups who have tried on a limited basis to amass such programs.

ZX-81/TS1000 Tips: Instead of PAUSE 200 try PAUSE CODE "COS" to save memory. When making a copy of your program on tape (without data) redimension arrays, by for example DIM A\$(0,0) to save memory and loading/saving time, then on program execution, redimension them properly to restart the program. A good place to place the DIM statement (or GOSUB to more than one DIM statement) to redimension the arrays properly is line 0. In line zero, a RUN statement will redimension the array and clear it (the RUN would clear it anyway) but a GOTO 1 (used to restart without clearing array data) won't call the redimensioning (which would clear the data).



Published-Program Survey of Newsletters: Dominoes-ZX-81 m/c, BASIC line trace ZX-81, Vector Screen Calculator ZX-81 BASIC all in Dec.89-Vancouver TSUG N/L.....  
...octal/decimal conversion ZX-81 m/c, Sincus, Johnson City, NY N/L. (Nov-Dec,89)..  
....m/c utilities to break any program caught in a loop, to ignore BREAK command, invert screen attributes, scroll the screen line to left and wrap it around to next line, to flip the screen (horiz. or vert.) TS2068 m/c -Boston Comp. Soc., Sept-Oct n/1 (note: the TSUG there has recently broken off from Boston Comp. Soc.)  
...USA Medicare Income Tax TS2068 BASIC, in Harrisburg TSUG n/1 Sept.89....TS2068 State Lotto program, Aug & Sept,89 SMUG n/1, Milwaukee....crypto demo TS2068, BASIC and Chaos Demo, Oct.89 Vancouver TSUG N/L..... Cube-It (TS2068 game), Las Vegas, n/1 Jan., 1990....

An excellent series of assembly language articles with sample routines is being run these days in the Vancouver TSUG N/L. The articles, by V. Lee, are on Z-80 assembly language programming with the ZX-81/TS1000, but should also be a help to TS2068 users. Note that back issues of this newsletter also have articles by Wilf Rigger on m/c routines to speed up the ZX-81/TS1000 by modifying the display routines. Apparently using these techniques, the ZX-81/TS1000 can be made to run faster than by merely putting it into FAST mode. Seems like a much better way than brute force acceleration which only boosts execution 15% when you step up the ZX-81/TS1000 clock to 4MHz (from 3 $\frac{1}{2}$ MHz) and avoids the technical complications (like setting fire to your tv!). And of course good m/c or assembly language programming speeds up programs too. So can the rewriting of BASIC routines. An 8 sec. loop was cut down to about 1 $\frac{1}{2}$  sec. in one of my programs by ingenious rewriting of it to avoid going through the loop as many times and replacing looping with straight through BASIC code (within a loop that then required fewer times through the looping).

N/L Hardware articles continue to appear although the Z-88 cable articles<sup>(a)</sup> in HATS) and the increasing attention given to QL modifications, are starting to replace ZX-81/TS1000/TS2068 modifications. The writer hopes to print a lot of tips for hardware modifications of the ZX-81/TS1000 in the next pages of the 'Tips, Tricks and Techniques from the User Group Masters for the ZX-81/TS1000' booklet. This should become an excellent source of info and references for both the old and new ZX-81/TS1000 user. (Yes, new users are coming to the user groups looking for ZX-81/TS1000 lore. They are mainly young students seeking a low cost computer to practice the things they are being taught in school about computers since the hands-on practice sessions at school are often not quite long enough for them.) The writer thinks that not only is the ZX-81/TS1000 a good platform for controller projects like burglar alarm monitors but also a natural as the intelligent part of eprom or other programmable chip programmers. One user has made a short ZX-81/TS1000 board, with only the ROM, RAM and eprom (no ULA, keyboard or video) so that programs debugged and developed on the full ZX-81/TS1000 can be used on the short board, run from an eprom, rather than a full ZX-81/TS1000. It's been it is said by Donald Lambert of the CRAGIST, used to take pictures from a kite. So there is lots of potential left for hardware hackers in the old ZX-81/TS1000. Tip from Donald Lambert for TS2068 LOAD problems caused by heat: take off all unnecessary accessories off the bus/expansion. Also mentioned is an audio xformer tip passed on from Dave Solly, Ottawa, for this cassette load problem. (Sincus n/1, Dec.89)

Aerco TS2068 disk system users may be interested in the LKDOS Cartridge i/f for that system. With an FD-68 it gives Larken DOS & Extended BASIC commands incl. MERGE. You can switch between the Aerco & LKDOS so you can still use RP/M. It uses the Aerco RAM to give you a 48K RAM disk. Cartridge is also available for Oliger system & Ramex. US\$68 from Larken Electronics, RR#2, Navan, Ont., Canada K4B 1H9

Nazir Pashtoon of SMUG reports Cambridge North Am. in Chapt.11-but Sharps has Z-88 still and will now be only importer on the bankruptcy of Cambridge. (Milwaukee)

A software interface for the video digitizer for the TS2068 and the Okimate 20 dmp sold through TOYS-R-US for \$99, has been produced by John McMichael, 1710 Palmer Dr., Laramie, WY, USA 82070 for US\$20-requires IBM parallel Plug 'n Print cartridge. The product name is VideoCopy plus VideoTex. VideoTex alone avail. separately US\$10. This software uses the digitizer i/f board to capture VCR/videocamera images that

has been reviewed in past issues of SMUG Bytes, 5052 N. 91st St. Milwaukee, WI. In amateur publication of TS Bulletin, Bill Harmer, 97 Ruskin, Ottawa, Ont. Can. K1Y 4B3



-----  
American Micro Systems, 2175 Aborn Rd., Suite 262, San Jose, Calif., USA 95121 has a catalogue of QL (and Z-88) items, hardware and software, printed in TimeLinez. Included are QL:ROM upgrades, Trump Card disk i/f (\$312), IBM software emulator (\$172), modem adapter (\$69), Front Page desktop pub (\$34), CP/M emulator (Success), (\$69) Forth (\$57), Fortran 77(\$184), Pascal(\$57), C Language (\$69), Superbasic to C converter (needs C compiler)(\$80), Spellchecker (30,000 word)(\$68, needs extra RAM), DiscOver (disk file format converter)(\$46), Flight Sim.(\$34), Astrol.(\$35+) the QL computer itself (\$129), Lightning (accelerator software)(\$46), Serv.Man.(\$29) 30 meg. hard drive coming soon, JSU ROM (\$19), membrane keyboard(\$25) all in US\$ plus Z-88 products, the Z-88 \$450, PctoZ-88 link-up(\$70), QLtoZ-88(\$46), Mac to Z-88(\$111) zterm(\$90), zBASE(\$120), Superchip(\$200), Modem, 1200(\$130), 32Keprom(\$40) plus plus ++ their phone no. is (408)270-9730 (voice, ans. mach.)

.....  
Maxcom TS2068 BBS system for LKDOS disk users: "Maxcom is everything and more that it's chalked up to be...I've got to tell you, if you haven't gotten it as yet, chase those moths out of your wallet...Maxcom will truly make your day." from a column in the Las Vegas TSUG newsletter by Ralph Hammer. Also mentioned is the deluxe move utility in Maxcomm that allows transfer of files between disks, etc. simply and effectively. RMG handles this system in the USA. Nice to hear such good comments. Larry Kenny of Larken is also working on a desktop publisher with spelling checker at the moment (Jan., 1990) and is marketing a plotter, engraver, light duty plastic-router/miller that will run from an IBM PC and is also working on Z-80 controller version of it. The software to run it has been written in Hi-Soft Pascal on the TS2068. It will work on a 2'x2' board to a depth of 4". At an Ottawa-Hull TSUG user group several years ago, a TS2068 user demo'd a system in which a plotter applied etch-resist directly on a copper pcb for one-of-a-kind CAD/CAM production of pc boards, automatically. There is obviously lots that can still be done with the old Timex-Sinclair's. When the writer moves his old ZX-81 out of the computer den it will be going to the basement, electronics workshop for use in eprom blasting/controller experiments. These old models serve on, and on...

.....  
Fix for Timemachine BASIC compiler, change POKEs in line 106 from 32880 to 33880, by HL Schaaf printed in Aug.89, CATS & Dec., 1989 TimeLinez....

.....  
TDM Magazine, Vol.5, No.2 issued finally. contact TDM BBS at 503-244-2658 (8/1/n)

.....  
Jack Dohany has upgraded MScript to version 5.5 according to D.Lambert, CRAGIST

.....  
TS1000's still available-contact Gary Young, 586 9th Ave., Marion, IA, USA 52302-cheap working ones or bad keyboard ones avail., ram packs (16K) and progs. can be incl.

.....  
Article on using an audio transformer as a cassette load aid (TS1000/TS2068), in Aug-Dec., 1989 issue of Cragist, D. Lambert, 3310 Clover Dr. S.W., Cedar Rapids, IA 52404

.....  
Source code for TreeFORTH (a romable, TS1000, slightly non-standard FORTH) has been released for non-commercial use by author Bob Alsum. A TS1500 with Tree FORTH in ROM has turned up on the surplus market, D. Lambert, CRAGIST, reports.

.....  
Disk Library, TS2068 disk utilities program (search, list, delete, update, sort etc.) has been created by RA Hilsman, POB 45, Menomonee Falls, WI, USA 53051 and is avail. on cassette for US\$5 (or possibly slightly more mailed to Canada/for.). It is for the TS2068 Oliger system but probably can be adapted for other disk systems by the programmer/hacker. -----Mentioned in Sincus n/1 Jan.90 & SMUG n/1 Apr., 1988

.....  
Sincus N/L from 1229 Rhodes Rd., Johnson City, NY, USA 13790 continues to run lists of TS2068 ROM routines listed different ways, sorted and compared with Spectrum.

.....  
Don Lamen of Sincus continues to submit articles on TS1000 programming, mainly m/c to Sincus n/1 and it is nice to see the TS1000 still receiving support like that.

.....  
Computer Shopper has discontinued articles on Commodore 64, Sinclair, TI, Atari 8-bit much to our disappointment. Latter Sinclair articles were really good. Not much reason for me to buy it anymore except for hardware adverts re PC clones. User Gp. listings continue however & Sinclair Gps. will pop up listed there.

.....  
Published as an amateur venture by Bill Harmer, 97 Ruskin St., Ottawa, Ont., Canada K1Y 4B3. Don't hack, promise, send money or buy without checking info in this n/1 with a second, independently qualified source. You may photocopy freely for friends.

BOB MITCHELL  
302 CANTON COURT #56  
SUN CITY CENTER FL 33573  
DATE: 900225

Dear Les,

George Chambers sent me the latest Sinc-Link plus a copy of your NMI-F routine to get back to the Autostart file in the Ramdisk. Well, that's a dandy and I soon put it to use in my Omnibus file which is my Autostart. Enclosed is a copy of a letter I have just done to George and I thought you might like to read through the things I did.

My wife and I are spending the winter just south of Tampa in a place called Sun City Center where we have rented for the last four years. The weather is certainly hard to beat!

I was interested in your use of the RAMDISK to do letters etc while at lunch at your job at Kennedy Space Center. I see that you are using Mscript and that's what I use for most of my letters etc. It is much better than Tasword and is just about as good as any word processor you would want.

You do a lot with just two chips and I suppose you will add some more before long. George says there are two sorts of SRAM chips and that one kind has some sort of refresh circuit built in and that is the kind we should not use. The last four chips I got have given me a few CRC errors as I explain in my letter to him but I still feel that most of my problems are with the edge connectors. I have tried to clean them with some Tuner Cleaner from Radio Shack but I may not have them squeaky clean yet. Do you have any thoughts on this matter. My first four chips are 62256LP-15 while the last four piggy-backed onto the first are 62256ALP10/51257L10.

Your use of a lithium battery is interesting; George did the work putting on the sockets on my board and found he had to raise the voltage so he added a third AAA battery. Larry Kenny was a bit surprised but said not to go any higher than three.

Glad you are able to make use of the Tasort routine that I wrote some time ago. I use it quite a bit and while I have a similar program for Mscript, I usually use Tasword for lists since Tasort is easier to use. You would also be interested in my remarks about improving my index program; that is a long term project, I expect. It all depends on whether Larry Crawford can come up with a suitable DELETE routine.

By the way, if you are interested in getting into bank-switching, I can send you what I have. With two chips you can handle an awful lot of data.

What are you going to do for 2040 paper now that Radio Shack in the U.S. are phasing theirs out.

Well, I'll close this off. One final word: we Survivors are now finding that the clubs are the only way left to be able to expand our use of the 2068 and from my experience, the horizon never seems to get any closer.

Best regards,

Bob.



BOB MITCHELL  
302 CANTON COURT #56  
SUN CITY CENTER FL 33573  
DATE: 900226

Dear George,

Received your letter along with the delayed Sinc-Link and the copy of the m/c routine to put an NMI-F autoloader feature into a menu. I did some work on the latter. I enclose the results of my work. Here's what I've done. I typed in the code you sent me and loaded it temporarily into addresses 56100 to 56164. Then I used the BASIC routine to POKE it into the area in LKDOS 16310 and up. Tried it and of course it did not work. So I loaded Monitor and took a look at the disassembly and spotted the problem at address 56115/6. Had to change the LD HL to 16365 to go with starting the code at 16310. This so it would not conflict with other code that I park there for Graphics printing and other things.

Then I decided to load the code from my disk from 56100 to the free area in 2068 RAM (which starts at 24311) and I used the area 24495 to 24560. I have other stuff parked in other parts of this free area. The beauty of this is that once the code is loaded into this free area it gets saved with an Autostart save and is there for keeps. It will not be saved with a standard SAVE. However, it will always be necessary to POKE it into the LKDOS RAM on power up. I think the delay once on power up is tolerable. However, an LDIR could be added but I leave that up to you.

Now the next thing was to stop the poking action after it had been done once at power up. I did this by finding the value of address 16310 (line 11) and checking whether it was 123 (line 32). The value 123 is the value of the first address in the code. Finally, just in case the LKDOS address 8214 gets changed, I put in a line (41) to restore the value to 16310. So take a look at the sheet enclosed and I think that I've covered everything.

Checked a lot of programs and NMI-F worked fine except the following:

Timachine: had to use RAND USR 0 then NMI-F;

Spectrum Emulator: had to use OUT 244,0: OUT 7,B then NMI-F.

I see that I sent some input to you for Sinc-link done on Pixel Print Press which had some definitions which I culled from a new Computer Dictionary I bought down here. So there is no panic to write anything for the next issue.

Larry Crawford and I are corresponding on the bank-switching exercise and gradually getting it smoothed out. We still need a good DELETE routine so that I can use the data-base for my Disk Indexer and so that it will hold more than the current 1400 record limit. Besides I want to put in a few extra bytes in each record to allow for a brief description of the record, seeing that the file names sometimes get pretty cryptic.

I am still getting the odd CRC error on the RAMdisk but they are rather rare and only seem to happen when I am experimenting; then they can appear on any tracks on any of the chips (but never the first four chips), usually only one or two tracks but sometimes as many as the whole six of one chip. The write protect switch is no deterrent. The CRC errors never seem to appear when I am saving to RAMdisk or on power up. I am puzzled but it only takes a few moments to load in the RAMdisk from a disk so it's not much of a problem. However, I've written to Jameco about it.

Re Ariel Fralich, what happened to the rights to Timachine? Does Cameron Hayne still have a copyright on it? And where is Cameron anyway? I'd like to know why we can't get a COMPLETE copy of the output of Timachine on the wide printer. He might have a fix for that.

I'm off to Tampa on Tuesday and will pick up the daisywheel during that trip.

I'll send a copy of all this to Les Cottrell in Cocoa and thank him for doing the routine.

Best regards,

Bob.

To Les Cottrell .